Applying clique-decomposition for computing Gromov hyperbolicity

Nathann Cohen, David Coudert, Guillaume Ducoffe, Aurélien Lancin

Applying clique-decomposition for computing Gromov hyperbolicity*

Nathann Cohen[1], David Coudert[2], Guillaume Ducoffe[2], and Aurélien Lancin[3]

[1]CNRS, Univ Paris Sud, Université Paris-Saclay, France
[2]Université Côte d'Azur, Inria, CNRS, I3S, France
[3]LAMIA Laboratory, University of the French West Indies, France

Abstract

Given a graph, its *hyperbolicity* is a measure of how close its distance distribution is to the one of a tree. This parameter has gained recent attention in the analysis of some graph algorithms and the classification of complex networks. We study on practical improvements for the computation of hyperbolicity in large graphs. Precisely, we investigate on relations between the hyperbolicity of a graph G and the hyperbolicity of its *atoms*, that are the subgraphs output by the clique-decomposition invented by Tarjan [51, 65]. We prove that the maximum hyperbolicity taken over the atoms is at most one unit off from the hyperbolicity of G and the bound is sharp. We also give an algorithm to slightly modify the atoms, called the "substitution method", which is at no extra cost than computing the clique-decomposition, and so that the maximum hyperbolicity taken over the resulting graphs is *exactly* the hyperbolicity of the input graph G. An experimental evaluation of our method for computing the hyperbolicity of a given graph from its atoms is provided for collaboration networks and biological networks. Finally, on a more theoretical side, we deduce from our results the first *linear-time* algorithm for computing the hyperbolicity of an outerplanar graph.

Keywords: Gromov hyperbolicity; graph algorithms; clique-decomposition; outerplanar graphs

1 Introduction

In this paper we aim at improving the computation of hyperbolicity in graphs whose size ranges from thousands to tens of thousands of nodes. To this end, we establish new relations between hyperbolicity and some graph decomposition. Roughly, the hyperbolicity of a metric space is an estimate of how close it is to a metric tree (formal definitions are postponed to the technical sections of this paper). This parameter was first introduced by Gromov in the context of automatic groups [41][1]. Later on, it was applied to the study of more general metric spaces including graphs equipped with their shortest-path metric. Graph hyperbolicity is now part of the parameters in use to classify complex networks [1, 3, 47]. In particular, it has been proposed

*This work is partially supported by ANR project Stint under reference ANR-13-BS02-0007 and ANR program "Investments for the Future" under reference ANR-11-LABX-0031-01.

[1]Informally speaking, elements of an automatic group are the vertices of some (Cayley) graph and it can be checked with finite-state automata whether two words represent either a same vertex or two adjacent vertices. Formal definitions and properties of automatic groups are out of our scope. We refer the reader to [36] for more information.

in [14] to consider hyperbolicity as a measure of how much a network is "democratic", with the level of "democracy" in a network growing with the minimum size of a *core* (subset of vertices intersecting a constant fraction of all the shortest-paths). Experiments in [2, 14] suggest that the larger the hyperbolicity of a real-world graph, the more democratic it is (see also [23] for formal relationships between Gromov hyperbolicity and the existence of a core). Furthermore, the study of graphs with bounded hyperbolicity has found applications in the design and the analysis of approximation algorithms [22, 29] and geometric routing schemes [12], as well as in network security [45] and bioinformatics [20, 35], to name a few. As a result, hyperbolicity and its relations to other metric graph parameters has received growing attention over the last decades. The reader may refer to [1, 34] for a recent survey.

Computing the hyperbolicity is useful in some of the above applications. For instance, it allows to compute an embedding of the graph into the hyperbolic plane with quasi-optimal distortion of the distances in linear time [67]. The latter is a prerequisite to many algorithms on negatively curved spaces [49]. However, the computational cost of the hyperbolicity has only recently received a bit more attention. So far, the best-known algorithm to compute the hyperbolicity [37], though it runs in polynomial-time, is impractical for large-scale graphs such as the graph of the Autonomous Systems of the Internet, road maps, etc. This comes from its challenging implementation, relying on fast square matrix multiplications, and its time complexity which is supercubic. On a more positive side, there have been recent attempts that are much more efficient than the state-of-the-art algorithm in practice, with running time dominated by the computation of the all-pairs-shortest-paths [15, 26]. But on the negative side, a bunch of complexity assumptions – including the Strong Exponential Time Hypothesis [44] – implies that graph hyperbolicity cannot be computed in subquadratic-time, even for sparse graphs [16, 27, 37]. This motivates us to study which structural properties can help to speed-up the computation of the hyperbolicity in large graphs.

Our approach We study how the information provided by a *clique-decomposition* of a graph into *atoms* (see Section 2.2) can be used to compute the hyperbolicity. As such a decomposition is readily implemented, we strive to split the problem into the computation of the hyperbolicity of smaller graphs similar to the *atoms* themselves.

Note that there are studies supporting the existence of clique-separators in real-life graphs, such as the underlying graphs of social and biological networks [1, 8, 31, 46], that makes our approach practical for large graphs. Furthermore, clique-decomposition has been proved useful to preprocess the graphs in the computation of many optimization problems [65] – including the computation of treelength [33] – related to hyperbolicity. Therefore, at first glance, it is not surprising that clique-decomposition can be applied to preprocess the graphs in the computation of the hyperbolicity. This being said, hyperbolicity is less robust than other metric invariants to graph modifications (e.g. it may increase through an edge-contraction [19]), and so a careful analysis is needed to prove that it is indeed the case.

Our approach in this paper has similarities with the work presented in the PHD Thesis of Soto [59]: he proved for instance that the hyperbolicity of a graph is the maximum hyperbolicity taken over the subgraphs from the split-decomposition [28] or from the modular decomposition [39]. The latter has practical applications in protein-protein interaction networks [38].

Main contributions Our first result on clique-decomposition is in sharp contrast with those obtained in [59] for modular and split decompositions. Indeed, we prove that the hyperbolicity $\delta(G)$ of a graph G *cannot* be deduced directly from the hyperbolicity of its atoms (Section 3). We prove nonetheless that it can be approximated with additive constant 1 by taking the

maximum hyperbolicity $\delta^*(G)$ over the atoms (Section 4). This result requires an in-depth analysis of clique-decomposition in order to be proved. Additionally, we present necessary conditions for having that $\delta(G) > \delta^*(G)$.

Based on this characterization, we show in Section 5 how each atom can be transformed (i.e. augmented with few simplicial vertices) in order to compute exactly the hyperbolicity, and provide a complexity analysis of the procedure. Experiments in Section 7 show the benefit of our method in terms of size of the graph, when applied to some real networks from scientific communities and biology.

Finally, we apply clique-decomposition for improving the best-known complexity to compute the hyperbolicity in the class of outerplanar graphs. We detail in Section 6 the first linear-time algorithm for computing the hyperbolicity of these graphs. We find the latter result all the more interesting that under the Strong Exponential Time Hypothesis, the hyperbolicity of sparse graphs cannot be computed in subquadratic-time [16].

Definitions and notations used in this paper are introduced in Section 2.

2 Definitions and notations

We use the graph terminology of [13, 32]. All graphs considered in this paper are finite, unweighted and simple. Given $G = (V, E)$, let $n = |V|$, $m = |E|$. The open neighborhood $N_G(S)$ of a set $S \subseteq V$ consists of all vertices in $V \setminus S$ with at least one neighbor in S. The closed neighborhood of S is the set $N_G[S] = S \cup N_G(S)$.

Given two vertices u and v, a *uv-path* of length $l \geq 0$ is a sequence of vertices $(u = v_0 v_1 \dots v_l = v)$, such that $\{v_i, v_{i+1}\}$ is an edge for every i. In particular, a graph G is *connected* if there exists a uv-path for all pairs $u, v \in V$, and in such a case the *distance* $\mathrm{d}_G(u, v)$ is defined as the minimum length of a uv-path in G. Note that it yields a discrete metric space (V, d_G), also known as the shortest-path metric space of G. We also denote by $\mathrm{d}(u, X) = \min_{x \in X} \mathrm{d}(u, x)$ the distance between a vertex u and a set X of vertices.

Our proofs use the notions of subgraphs, *induced* subgraphs, as well as *isometric subgraphs*, the latter denoting a subgraph H of a graph G such that $\mathrm{d}_H(u, v) = \mathrm{d}_G(u, v)$ for any two vertices $u, v \in H$.

When G is clear from the context, we write d (resp. N) instead of d_G (resp. N_G).

2.1 Gromov hyperbolicity

The space (V, d_G) is a tree metric if there exists a distance-preserving mapping from V to the nodes of an edge-weighted tree. In this case, the graph G is called 0-hyperbolic. Several characterizations exist for 0-hyperbolic graphs. Informally, a graph is called δ-hyperbolic if it satisfies one of these characterizations up to a defect at most δ. Different characterizations lead to different values for δ, but they may differ only by a small constant factor [7, 30, 41]. We here consider the following 4-point definition for hyperbolicity.

Definition 1 (**4-points Condition, [41]**). *Let G be a connected graph. For every 4-tuple u, v, x, y of vertices of G, we define $\delta(u, v, x, y)$ as half of the difference between the two largest sums among*

$$S_1 = \mathrm{d}(u, v) + \mathrm{d}(x, y),\ S_2 = \mathrm{d}(u, x) + \mathrm{d}(v, y) \text{ and } S_3 = \mathrm{d}(u, y) + \mathrm{d}(v, x).$$

The hyperbolicity of G, denoted by $\delta(G)$, is equal to $\max_{u,v,x,y \in V(G)} \delta(u, v, x, y)$. Moreover, we say that G is δ-hyperbolic *whenever $\delta \geq \delta(G)$.*

It is straightforward, by the above definition, to compute graph hyperbolicity in $\Theta(n^4)$-time. In theory, it can be decreased to $\mathcal{O}(n^{3.69})$ by using a clever $(\max, \min)$ matrix product [37]; however, in practice, the best-known algorithms still run in $\mathcal{O}(n^4)$-time [15, 26]. Graphs with small hyperbolicity can be recognized faster. In fact, 0-hyperbolic graphs coincide with *block graphs*, that are graphs whose all biconnected components are complete subgraphs [6, 43]. Hence it can be decided in linear $\mathcal{O}(n+m)$-time whether a graph is 0-hyperbolic. The latter characterization of 0-hyperbolic graphs follows from a more general result saying that the hyperbolicity of a graph is the maximum hyperbolicity of its biconnected components (our work give a new proof of this well-known result). More recently, it was proved that the recognition of $\frac{1}{2}$-hyperbolic graphs is computationally equivalent to decide whether there is a chordless cycle of length 4 in a graph [27]. The latter problem can be solved in deterministic $\mathcal{O}(n^{3.26})$-time [50] and in randomized $\mathcal{O}(n^{2.373})$-time [66] by using fast matrix multiplication.

2.2 Clique-decomposition

The clique-decomposition was introduced by Tarjan in [65] then made unique by Leimer in [51]. Given $G = (V, E)$, we name *separator* a subset of vertices $X \subset V$ such that the removal of X disconnects the graph. We call X a *clique-separator* when the induced subgraph $G[X]$ is a complete graph. A graph is *prime* if it does not contain a clique-separator. Examples of prime graphs are complete graphs and cycles. Finally, the *clique-decomposition* of G is the collection of its maximal sets of vertices that induce prime subgraphs of G (we call them *atoms*). See Figure 1 for an illustration. The decomposition is unique and it can be computed in $\mathcal{O}(nm)$-time [51, 65]. We refer to [9] for a survey on clique-decomposition.

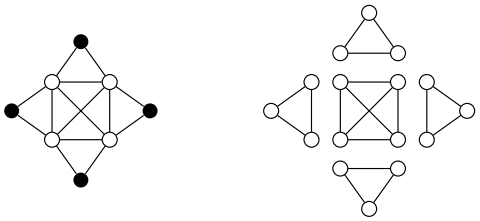


Figure 1: clique-decomposition of a graph with five atoms. A 4-tuple with hyperbolicity 1 appears in bold.

Notations Let us fix some notations for the proofs. Given $G = (V, E)$, let X be a separator of G. Let A, B denote two sets of vertices such that $A \cap B \subseteq X$ and both $A \setminus X$ and $B \setminus X$ are nonempty. We call X a $(A|B)$-separator. Let us denote by $(a|b_1, b_2, b_3)$ a 4-tuple such that $a \in A$ and $b_1, b_2, b_3 \in B$. In the same way, let us denote by $(a_1, a_2|b_1, b_2)$ a 4-tuple such that $a_1, a_2 \in A$ and $b_1, b_2 \in B$. Note that we allow some vertices of the 4-tuple to be in X with this notation.

3 Hyperbolicity and clique-separators

It may happen that every atom of a graph G is δ-hyperbolic while G has hyperbolicity strictly greater than δ. As an example, consider the chordal graph of Figure 1. It is 1-hyperbolic, with a 4-tuple of maximum hyperbolicity being drawn in black. However, its five atoms are complete graphs which are thus 0-hyperbolic.

The purpose of the next two sections is to upper-bound the gap between $\delta(G)$ and $\delta^*(G)$, where $\delta^*(G)$ denotes the maximum hyperbolicity of the atoms of G. To this end, we analyze in

this section the relationship between the hyperbolicity of a graph and a given clique-separator, leading to the approximation with additive constant of Theorem 12. It begins with an observation about $(a_1, a_2|b_1, b_2)$ 4-tuples and the diameter $\mathrm{diam}(X) = \max_{u,v \in X} \mathrm{d}_G(u, v)$ of a $(A|B)$-separator X.

3.1 Hyperbolicity of $(a_1, a_2|b_1, b_2)$ 4-tuples

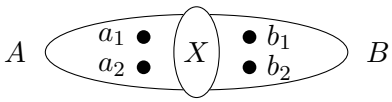


Figure 2: Illustration of a $(A|B)$-separator.

Lemma 2. *Let X be a $(A|B)$-separator of a connected graph G. For every $(a_1, a_2|b_1, b_2)$ 4-tuple, we have $\delta(a_1, a_2, b_1, b_2) \leq \mathrm{diam}(X)$.*

Proof. By Definition 1, we have $\delta(a_1, a_2, b_1, b_2) = (L - M)/2$ where L and M are the two biggest sums among the following:

$$S_1 = \mathrm{d}(a_1, a_2) + \mathrm{d}(b_1, b_2),\ S_2 = \mathrm{d}(a_1, b_1) + \mathrm{d}(a_2, b_2),\ S_3 = \mathrm{d}(a_1, b_2) + \mathrm{d}(a_2, b_1).$$

Let us upper-bound L. By the triangular inequality we have that for every $u, v \in \{a_1, a_2, b_1, b_2\}$, $\mathrm{d}(u, v) \leq \mathrm{d}(u, X) + \mathrm{d}(v, X) + diam(X)$. Thus, for every $i \in \{1, 2, 3\}$ we have (by applying twice the triangular inequality) that $S_i \leq \mathrm{d}(a_1, X) + \mathrm{d}(a_2, X) + \mathrm{d}(b_1, X) + \mathrm{d}(b_2, X) + 2 \cdot diam(X)$. In particular, $L \leq \mathrm{d}(a_1, X) + \mathrm{d}(a_2, X) + \mathrm{d}(b_1, X) + \mathrm{d}(b_2, X) + 2 \cdot diam(X)$.

Furthermore, since X is assumed to be a $(A|B)$-separator, we have that for every $i, j \in \{1, 2\}$, all $a_i b_j$-paths in G must intersect X, and so, $\mathrm{d}(a_i, b_j) \geq \mathrm{d}(a_i, X) + \mathrm{d}(b_j, X)$. Hence, $\mathrm{d}(a_1, X) + \mathrm{d}(a_2, X) + \mathrm{d}(b_1, X) + \mathrm{d}(b_2, X) \leq \mathrm{d}(a_1, b_1) + \mathrm{d}(a_2, b_2) = S_2$, and in the same way $\mathrm{d}(a_1, X) + \mathrm{d}(a_2, X) + \mathrm{d}(b_1, X) + \mathrm{d}(b_2, X) \leq S_3$. Altogether it implies that $L \leq \min\{S_2, S_3\} + 2 \cdot diam(X)$. By noticing that $\min\{S_2, S_3\} \leq M$, one finally obtains that $L \leq M + 2 \cdot diam(X)$, and so, $\delta(a_1, a_2, b_1, b_2) \leq diam(X)$, as desired. □

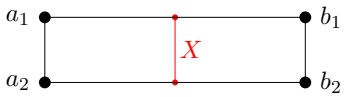


Figure 3: An $(a_1, a_2|b_1, b_2)$ 4-tuple with hyperbolicity 1.

Corollary 3. *Let X be a $(A|B)$-clique-separator of a connected graph G. For every $(a_1, a_2|b_1, b_2)$ 4-tuple, we have $\delta(a_1, a_2, b_1, b_2) \leq 1$.*

The upper-bound of Corollary 3 is sharp, as shown with the grid of Figure 3. By taking larger grids, it can also be shown that the upper-bound of Lemma 2 is sharp.

3.2 Hyperbolicity of $(a|b_1, b_2, b_3)$ 4-tuples

In contrast to $(a_1, a_2|b_1, b_2)$ 4-tuples, the hyperbolicity of a $(a|b_1, b_2, b_3)$ 4-tuple can be arbitrarily large. In what follows, we relate $(a|b_1, b_2, b_3)$ 4-tuples with some 4-tuples of $B \cup X$ in order to upper-bound their hyperbolicity. Precisely, note that X being a clique, each vertex $a \in A$ is at distance at least $\mathrm{d}(a, X)$ and at most $\mathrm{d}(a, X) + 1$ from any vertex of X. We now show how this can be used with respect to the hyperbolicity.

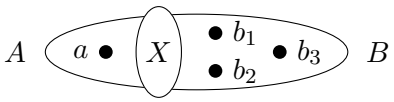


Figure 4: Illustration of a $(a|b_1, b_2, b_3)$-separator.

Lemma 4. *Let X be a $(A|B)$-clique-separator of a connected graph G. Given a $(a|b_1, b_2, b_3)$ 4-tuple, let $x \in X$ be such that $\mathrm{d}(a, x) = \mathrm{d}(a, X)$. We have $\delta(a, b_1, b_2, b_3) \leq \delta(x, b_1, b_2, b_3) + 1/2$.*

Proof. Let us assume w.l.o.g. that $d(a, b_1) + d(b_2, b_3) \geq d(a, b_2) + d(b_1, b_3) \geq \mathrm{d}(a, b_3) + d(b_1, b_2)$. We claim that $\mathrm{d}(x, b_1) + \mathrm{d}(b_2, b_3) \leq \max\{\mathrm{d}(x, b_2) + \mathrm{d}(b_1, b_3),\ \mathrm{d}(x, b_3) + \mathrm{d}(b_1, b_2)\} + 2 \cdot \delta(x, b_1, b_2, b_3)$. Indeed, the latter inequality is trivial if $\mathrm{d}(x, b_1) + \mathrm{d}(b_2, b_3) \leq \max\{\mathrm{d}(x, b_2) + \mathrm{d}(b_1, b_3),\ \mathrm{d}(x, b_3) + \mathrm{d}(b_1, b_2)\}$, otherwise it directly follows from the definition of $\delta(x, b_1, b_2, b_3)$ (cf. Definition 1). In this situation:

$$
\begin{aligned}
2 \cdot \delta(x, b_1, b_2, b_3) &\geq \mathrm{d}(x, b_1) + \mathrm{d}(b_2, b_3) \\
&\quad - \max\{\mathrm{d}(x, b_2) + \mathrm{d}(b_1, b_3),\ \mathrm{d}(x, b_3) + \mathrm{d}(b_1, b_2)\} \\
&\geq \big[\mathrm{d}(a, b_1) - \mathrm{d}(a, x)\big] + \mathrm{d}(b_2, b_3) \\
&\quad - \max\{\mathrm{d}(x, b_2) + \mathrm{d}(b_1, b_3),\ \mathrm{d}(x, b_3) + \mathrm{d}(b_1, b_2)\} \\
&\geq \mathrm{d}(a, b_1) + \mathrm{d}(b_2, b_3) \\
&\quad - \max\{\big[\mathrm{d}(a, x) + \mathrm{d}(x, b_2)\big] + \mathrm{d}(b_1, b_3),\ \big[\mathrm{d}(a, x) + \mathrm{d}(x, b_3)\big] + \mathrm{d}(b_1, b_2)\} \\
&\geq \mathrm{d}(a, b_1) + \mathrm{d}(b_2, b_3) \\
&\quad - \max\{\big[\mathrm{d}(a, X) + \mathrm{d}(b_2, X) + \mathit{diam}(X)\big] + \mathrm{d}(b_1, b_3), \\
&\qquad\qquad \big[\mathrm{d}(a, X) + \mathrm{d}(b_3, X) + \mathit{diam}(X)\big] + \mathrm{d}(b_1, b_2)\} \\
&\geq \mathrm{d}(a, b_1) + \mathrm{d}(b_2, b_3) \\
&\quad - \max\{\big[\mathrm{d}(a, X) + \mathrm{d}(b_2, X)\big] + \mathrm{d}(b_1, b_3), \\
&\qquad\qquad \big[\mathrm{d}(a, X) + \mathrm{d}(b_3, X)\big] + \mathrm{d}(b_1, b_2)\} - \mathit{diam}(X) \\
&\geq \mathrm{d}(a, b_1) + \mathrm{d}(b_2, b_3) \\
&\quad - \max\{\mathrm{d}(a, b_2) + \mathrm{d}(b_1, b_3),\ \mathrm{d}(a, b_3) + \mathrm{d}(b_1, b_2)\} - 1 \\
&\geq 2 \cdot \delta(a, b_1, b_2, b_3) - 1.
\end{aligned}
$$

Hence, $\delta(a, b_1, b_2, b_3) \leq \delta(x, b_1, b_2, b_3) + 1/2$. □

3.3 Disconnection by a clique-separator

Summing up the two previous Sections 3.1 and 3.2, we relate the hyperbolicity of a graph G with the hyperbolicity of the subgraphs obtained by disconnecting G with a clique-separator. The following Theorem 5 was proved independently in [59, 70] for separators of any diameter, and we prove it here for self-containment. Lemmas 2 and 4 will be reused in the following sections.

Theorem 5. *Let X be a clique-separator of a connected graph G, and let $C_1, \ldots, C_l$ be the connected components of $G \backslash X$. We define $G_i = G[C_i \cup X]$. We have :*

$$\max\{\delta(G_1), \ldots \delta(G_l)\} \leq \delta(G) \leq \max\{1/2, \delta(G_1), \ldots \delta(G_l)\} + 1/2.$$

Proof. Since X is a clique, and so, $G[X]$ is an isometric subgraph, every subgraph G_i is isometric as well. Hence, the lower-bound follows from the 4-point definition (Definition 1).

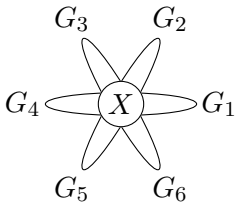


Figure 5: Illustration of a clique-separator X with the connected components of $G \setminus X$.

Let us now prove that $\delta(a,b,c,d) \leq \max\{1/2, \delta(G_1), \dots \delta(G_l)\} + 1/2$ holds for any $a,b,c,d \in V$. We consider a connected component C_i minimizing the number of vertices in the 4-tuple a,b,c,d that are *not* in the block $C_i \cup X$. There are three cases to be considered.

- If $a,b,c,d \in C_i \cup X$ we are done as $\delta(a,b,c,d) \leq \delta(G_i)$.
- If all of a,b,c,d but one vertex are in $C_i \cup X$ let us assume w.l.o.g. that $a \notin C_i \cup X$. Then a,b,c,d is a $(a|b_1,b_2,b_3)$ 4-tuple, for the choices of $B = C_i \cup X$ and $A = V \setminus C_i$. By Lemma 4 it follows that $\delta(a,b,c,d) \leq \delta(G_i) + 1/2$.
- Else, there are no more than two vertices among a,b,c,d that are in $C_i \cup X$. Suppose w.l.o.g. $a,c \notin C_i \cup X$. Let j,k satisfy $a \in C_j$ and $c \in C_k$. By minimality of $|\{a,b,c,d\} \setminus (C_i \cup X)|$ we have $b,d \notin C_j \cup C_k$. Therefore, a,b,c,d is a $(a_1,a_2|b_1,b_2)$ 4-tuple, for the choices of $A = C_j \cup C_k \cup X$ and $B = V \setminus (C_j \cup C_k)$, and we conclude by Corollary 3 that $\delta(a_1,a_2,b_1,b_2) \leq 1$ in this case.

□

The upper-bound of Theorem 5 is sharp. It can be shown using the graph in Figure 6, constructed from a cycle C_7 of length 7 to which we add a triangle.

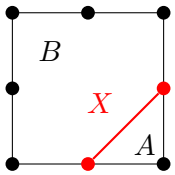


Figure 6: X is a $(A|B)$-clique-separator: we have $\delta(G) = 3/2$, while $\delta(G[B]) = 1$, and $\delta(G[A]) = 0$.

4 Hyperbolicity and clique-decomposition

In Section 3, we gave a sharp upper-bound on the distortion of hyperbolicity when the graph is disconnected by a single clique-separator. The atoms of the graph result from its disconnection by some clique-separators [51]. However, Theorem 5 does not apply to a whole clique-decomposition as the successive approximations would add up. We thus need to find additional properties to approximate the hyperbolicity of a graph from computations on its atoms in order to prove Theorem 12.

Our proofs in this section are based on the property that the atoms of a graph can be organized into a tree (sometimes called an atom tree [10] or a maximal prime subgraph junction tree [55]). Using this tree, any 4-tuple with large hyperbolicity can be related to an atom that is most "central" to it (this will be made more precise in the following). We can then upper-bound

the difference between the hyperbolicity of the 4-tuple and the hyperbolicity of this atom. For the latter, a delicate technical argument is needed in order to obtain the sharp upper-bound on the difference.

4.1 Relating atoms and 4-tuples with large hyperbolicity

We aim at relating every 4-tuple a, b, c, d with a sufficiently large hyperbolicity to some atom by which all the paths between a, b, c, d go through. The difference between $\delta(a, b, c, d)$ and the hyperbolicity of this atom will be studied next. Our result in this section involves basic knowledge of *tree-decomposition* (see [11]). A *tree-decomposition* $(T, \mathcal{X})$ of a graph $G = (V, E)$ is a pair consisting of a tree T and of a family $\mathcal{X} = (X_t)_{t \in V(T)}$ of subsets of V indexed by the nodes of T and satisfying:

- $\bigcup_{t \in V(T)} X_t = V$;
- for any edge $e = \{u, v\} \in E$, there exists $t \in V(T)$ such that $u, v \in X_t$;
- for any $v \in V$, $\{t \in V(T) \mid v \in X_t\}$ induces a subtree of T, denoted by T_v.

The sets X_t are called *the bags* of the decomposition. In the following, we use the property that there exists a tree-decomposition where the bags are exactly the atoms [10, 55].

Lemma 6. *Let $a, b, c, d \in V$ be a 4-tuple satisfying $\delta(a, b, c, d) \geq \frac{3}{2}$ in a connected graph $G = (V, E)$. There exists an atom A_0 of G such that $\forall u \in \{a, b, c, d\} \setminus A_0$, there is a clique-separator $X_u \subseteq A_0$ which separates u from $\{a, b, c, d\} \setminus \{u\}$.*

Proof. Let $(T, \mathcal{X})$ be a tree-decomposition of G whose bags are the atoms of G. Such a tree-decomposition was proved to exist in [10, 55]. In order to prove the lemma, we shall find an atom A_0 with the property that no more than two vertices among $\{a, b, c, d\} \setminus A_0$ are in the same connected component of $G \setminus A_0$. To find this atom, we weight the bags of $\mathcal{X}$ (we then choose the atom A_0 in the *weighted centroid* of T).

Precisely, for every of a, b, c, d we pick an atom which contains it and we define the weight of an atom as the number of times it has been picked. In particular, an atom has weight between 0 and 4, and the sum of weight of the atoms is equal to $\mathcal{W} = 4$. It is well-known that for any node-weighted tree with sum of weights $\mathcal{W}$, there is a node whose removal splits the tree into connected components where the sum of weight of the nodes is at most $\mathcal{W}/2$ [40]. So, let A_0 be an atom of G such that no component of $T \setminus \{A_0\}$ has the sum of weight of its bags greater than 2. We claim that $\forall u \in \{a, b, c, d\} \setminus A_0$, there is a clique-separator $X_u \subseteq A_0$ which separates u from $\{a, b, c, d\} \setminus \{u\}$, that will prove the lemma.

Indeed, let $u \in \{a, b, c, d\} \setminus A_0$ be arbitrary. By the properties of a tree-decomposition, T_u (induced by the atoms containing u) is the subtree of a component C_u of $T \setminus \{A_0\}$. Let $V_u \subseteq V$ be the subset of vertices that are contained in an atom in C_u, and let $A_u \in C_u$ be the atom that is adjacent to A_0 in T. Since A_u and A_0 are atoms of G, their intersection, denoted by $X_u = A_u \cap A_0$, is a clique [9]. Furthermore, by the properties of a tree-decomposition, X_u is a $(V_u | V \setminus V_u)$-separator of G. Therefore, we are left to prove that no vertex of $\{a, b, c, d\} \setminus \{u\}$ is in V_u, for the latter will prove that X_u is a clique-separator which separates u from $\{a, b, c, d\} \setminus \{u\}$. Assume for the sake of contradiction the existence of a vertex $v \in \{a, b, c, d\} \setminus \{u\}$ that is contained in V_u. We distinguish between two cases.

- Suppose that $v \notin X_u$. In this situation, T_u, T_v are subtrees of C_u. It implies that the sum of weight of the atoms in C_u is at least 2, and so, by the choice of atom A_0, it is equal to 2. In particular, u, v are the only two vertices of the 4-tuple that are in

$V_u \setminus X_u$ (else, the sum of weight of the atoms in C_u should be at least 3). Let $X = X_u,\ A = V_u,\ B = (V \setminus V_u) \cup X_u$. The 4-tuple a, b, c, d is a $(a_1, a_2|b_1, b_2)$ 4-tuple with $a_1 = u,\ a_2 = v$. Therefore, $\delta(a, b, c, d) \leq 1$ by Corollary 3, that contradicts the hypothesis that $\delta(a, b, c, d) \geq \frac{3}{2}$.

- Else, $v \in X_u$ and we can assume w.l.o.g. that no vertex of $\{a, b, c, d\} \setminus \{u\}$ is in $V_u \setminus X_u$ (else, we go back to the previous case). In this situation, let $X = X_u,\ A = V_u,\ B = (V \setminus V_u) \cup X_u$ as before, the 4-tuple a, b, c, d is a $(a_1, a_2|b_1, b_2)$ 4-tuple with $a_1 = u,\ a_2 = v$. Therefore, similarly as for the previous case, we have that $\delta(a, b, c, d) \leq 1$ by Corollary 3, that contradicts the hypothesis that $\delta(a, b, c, d) \geq \frac{3}{2}$.

As a result, no vertex of $\{a, b, c, d\} \setminus \{u\}$ is in V_u, and so, X_u is a clique-separator which separates u from $\{a, b, c, d\} \setminus \{u\}$. Since $X_u \subseteq A_0$, the latter proves the claim on A_0, hence the lemma. □

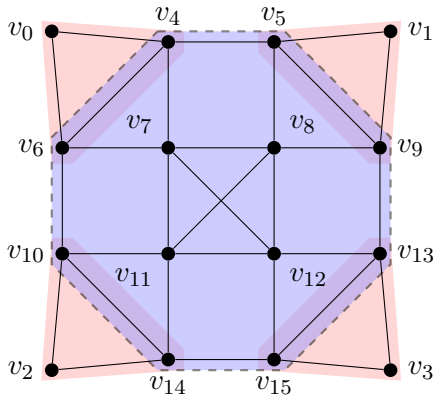


Figure 7: A 2-hyperbolic graph with five atoms: four are 0-hyperbolic, one is 1-hyperbolic. The 4-tuple v_0, v_1, v_2, v_3 has maximum hyperbolicity 2.

As an illustration, one may notice that the central atom in Figure 7 satisfies the property of Lemma 6 with respect to the 4-tuple v_0, v_1, v_2, v_3. Indeed, none of the four vertices is contained in this atom, but each of them is simplicial and can be separated from the three others by its two neighbors.

Discussion It is tempting to attempt a generalization of Lemma 6 to some other graph decompositions. In particular, we may wish to consider a decomposition of the graph by separators of diameter at most k, where k is a small constant (the case $k = 1$ corresponds to the clique-decomposition). If we assume in addition that the subgraphs are organized into a tree (*i.e.* they are the bags of a tree-decomposition), the generalization of Lemma 6 to that case is easy. However, in general there is no such tree-decomposition, which kills all the arguments in our proof.

We can go one step further and prove the following claim.

Proposition 7. *For every $G = (V, E)$, let $G_1, G_2, \ldots, G_l$ denote the isometric subgraphs of G that do not contain any isometric separator [2] of diameter at most two. The difference between $\delta(G)$ and $\max_i \delta(G_i)$ can be arbitrarily large.*

Proof. We shall prove that the result follows from some properties of *bridged* graphs. Indeed, we recall that a vertex is dominated if its closed neighborhood is included in the closed neighborhood

[2] We name a separator isometric when it induces an isometric subgraph.

of another vertex, and a graph is bridged if every of its isometric subgraphs of size at least two contains a dominated vertex [4]. We claim that for every bridged graph G, all the subgraphs G_i have hyperbolicity at most one. Since the hyperbolicity of bridged graphs can be arbitrarily large [48], the result follows.

In order to prove the claim, let G be a bridged graph and let G_i be an isometric subgraph of G where there is no isometric separator of diameter at most two. If G_i has order at most 1, then we are done as $\delta(G_i) = 0$ in this case. So, let us assume G_i contains at least two vertices. Since G is bridged, G_i contains a dominated vertex v_i. Let $u_i \in V(G_i)$ be a vertex dominating v_i. In this situation, u_i is universal in G_i, or else $N_G[u_i] \setminus v_i$ would be an isometric separator of G_i of diameter at most two. Therefore, G_i has diameter at most two, and so, $\delta(G_i) \leq 1$ [26], that proves the claim, hence the result. □

4.2 An additive approximation for hyperbolicity

From Lemma 6 we can associate a specific atom to a 4-tuple of large hyperbolicity. Four applications of Lemma 4 are then sufficient to prove that the hyperbolicity of this 4-tuple and the hyperbolicity of the atom differ by at most 2. The purpose of this section is to prove that this difference is in fact at most 1. To do this, we refine the results of Section 3.2.

We recall that Lemma 4 associates a $(a|b_1, b_2, b_3)$ 4-tuple to a 4-tuple x, b_1, b_2, b_3, with $x \in X$, such that $\delta(a, b_1, b_2, b_3) - \delta(x, b_1, b_2, b_3) \leq 1/2$. The difference between the hyperbolicity of the 4-tuples depends on the choice of x and on some properties of the $(a|b_1, b_2, b_3)$ 4-tuple. So, we first deepen our analysis of the worst-case when it is equal to $1/2$. We finally prove that when we apply this Lemma 4 twice on a 4-tuple with a large hyperbolicity, this maximum difference of $1/2$ can occur at most once.

For simplicity, we first reduce to the case when $\mathrm{d}(a, X) = 1$.

Lemma 8. *Let X be a $(A|B)$-clique-separator of a connected graph G, and let $a \in A$. We consider the graph G' obtained from G by adding a vertex a^* adjacent to $\{x \in X : \mathrm{d}_G(a, x) = \mathrm{d}_G(a, X)\}$. Then for every $b_1, b_2, b_3 \in B$ we have $\delta(a, b_1, b_2, b_3) = \delta(a^*, b_1, b_2, b_3)$.*

Proof. By construction G is an isometric subgraph of G' and so, $\forall\, u, v \in V(G)$, $\mathrm{d}_{G'}(u, v) = \mathrm{d}_G(u, v) = \mathrm{d}(u, v)$. In particular, the value $\delta(a, b_1, b_2, b_3)$ is not modified by the construction.

Let us relate $\mathrm{d}(a^*, b)$ with $\mathrm{d}(a, b)$ for every $b \in B$. Precisely, let us prove that $\mathrm{d}(a, b) - \mathrm{d}(a^*, b)$ is a constant (*i.e.*, not depending on b), that will prove by Definition 1 that $\delta(a, b_1, b_2, b_3) = \delta(a^*, b_1, b_2, b_3)$ for every $b_1, b_2, b_3 \in B$. In order to prove it, first observe that $\forall\, x \in X$, $\mathrm{d}(a, x) \in \{\mathrm{d}(a, X), \mathrm{d}(a, X)+1\}$ holds as X is a clique. Since a^* is adjacent to $\{x \in X : \mathrm{d}(a, x) = \mathrm{d}(a, X)\}$, this implies that $\forall\, x \in X$, $\mathrm{d}(a, x) = \mathrm{d}(a^*, x) + (\mathrm{d}(a, X) - 1)$. Furthermore, X is a $(A \cup \{a^*\}|B)$-separator of G'. Hence $\forall\, b \in B$, all a^*b-paths of G', resp. all ab-paths of G', intersect X. As a result, we have that for every $b \in B$, $\mathrm{d}(a, b) = \mathrm{d}(a^*, b) + \mathrm{d}(a, X) - 1$ and replacing a with a^* does not change the hyperbolicity of the 4-tuple a, b_1, b_2, b_3. □

Let us point out that adding a simplicial vertex in a graph G, as we do for the proof of Lemma 8, results in a new graph G' with potentially $\delta(G') > \delta(G)$. However the distances between the vertices of G are not modified (*i.e.*, G is an isometric subgraph of G'), that is all we need to ensure in order to compare the hyperbolicity of different 4-tuples of G. We come back to the difference between $\delta(G)$ and $\delta(G')$ in Section 5.

In what follows, we will need the following lemma, that is a technical generalization of Lemma 4.

Lemma 9. *Let X be a $(A|B)$-clique-separator of a connected graph G. Given a $(a|b_1, b_2, b_3)$ 4-tuple, write:*

$$S_1 = \mathrm{d}(a, b_1) + \mathrm{d}(b_2, b_3),\ S_2 = \mathrm{d}(a, b_2) + \mathrm{d}(b_1, b_3),\ S_3 = \mathrm{d}(a, b_3) + \mathrm{d}(b_1, b_2).$$

Assume w.l.o.g. that $S_1 \geq S_2 \geq S_3$, and let $x_2 \in X$ be such that $\mathrm{d}(a, b_2) = \mathrm{d}(a, x_2) + \mathrm{d}(x_2, b_2) = \mathrm{d}(a, X) + \mathrm{d}(x_2, b_2)$. If $\delta(a, b_1, b_2, b_3) > \delta(x_2, b_1, b_2, b_3)$, then we have:

- $S_1 > S_2 = S_3$.
- $\mathrm{d}(a, b_1) = \mathrm{d}(a, x_2) + \mathrm{d}(x_2, b_1)$.

Proof. For ease of calculation, we first reduce to the case when $\mathrm{d}(a, X) = 1$. Let G' be obtained from G by adding a vertex a^* with neighbors $\{x \in X \mid \mathrm{d}(a, x) = \mathrm{d}(a, X)\}$. By construction, G is an isometric subgraph of G', and so, the hyperbolicity of 4-tuples of $V(G)$ is not modified by the construction. Furthermore, by Lemma 8 we have $\delta(a^*, b_1, b_2, b_3) = \delta(a, b_1, b_2, b_3)$. In this situation, we can safely replace a with a^* in the 4-tuple, hence we may assume w.l.o.g. that $\mathrm{d}(a, X) = 1$ for the remaining of the proof.

For every i, let $x_i \in X$ denote a vertex on a shortest ab_i-path such that $\mathrm{d}(a, x_i) = \mathrm{d}(a, X) = 1$. In this situation, $\mathrm{d}(a, b_i) = \mathrm{d}(b_i, x_i)+1$. Let us introduce the indicator $\varepsilon_i = \mathrm{d}(x_2, b_i)-\mathrm{d}(x_i, b_i)$. We claim that $\varepsilon_i \in \{0, 1\}$, and $\varepsilon_i = 0$ if and only if x_2 is on a shortest ab_i-path. Indeed, since by the triangular inequality, $1+\mathrm{d}(x_i, b_i) = \mathrm{d}(a, b_i) \leq \mathrm{d}(a, x_2)+\mathrm{d}(x_2, b_i) = 1+\mathrm{d}(x_2, b_i)$, we have that $\varepsilon_i \geq 0$. Furthermore, since $x_2, x_i \in X$ and X is a clique, we also have by the triangular inequality that $\varepsilon_i \leq \mathrm{d}(x_2, x_i) \leq 1$. Altogether, $\varepsilon_i \in \{0, 1\}$, and we have $\varepsilon_i = 0$ if and only if $\mathrm{d}(a, b_i) = \mathrm{d}(b_i, x_2) + \mathrm{d}(a, x_2)$, that proves the claim. In particular $\varepsilon_2 = 0$.

Let us denote by S'_i the sum $\mathrm{d}(x_2, b_i) + \mathrm{d}(b_j, b_k)$, where $\{j, k\} = \{1, 2, 3\} \setminus \{i\}$. We aim to exhibit a relation between S_i and S'_i, that would yield in turn a relation between the values $\delta(a, b_1, b_2, b_3)$ and $\delta(x_2, b_1, b_2, b_3)$. At first we notice that $\mathrm{d}(a, b_i) = \mathrm{d}(x_i, b_i)+1 = \mathrm{d}(x_2, b_i)+1-\varepsilon_i$. So, we have:

$$\begin{aligned} S_i =& \mathrm{d}(a, b_i) + \mathrm{d}(b_j, b_k) \\ =& \mathrm{d}(x_2, b_i) + \mathrm{d}(b_j, b_k) + 1 - \varepsilon_i \\ =& S'_i + 1 - \varepsilon_i. \end{aligned}$$

Furthermore, by the hypothesis $S_1 \geq S_2 \geq S_3$ and $\delta(a, b_1, b_2, b_3) > 0$. Thus, by Definition 1 we have $S_1 = S_2 + 2\delta(a, b_1, b_2, b_3)$, and so, it holds that $S_1 > \max\{S_2, S_3\}$. The latter implies $S'_1 \geq S_1 - 1 \geq \max\{S_2, S_3\} \geq \max\{S'_2, S'_3\}$. More precisely:

- Suppose $S'_2 \geq S'_3$. In this situation, $\delta(a, b_1, b_2, b_3) = \frac{S_1-S_2}{2} = \frac{(S'_1+1-\varepsilon_1)-(S'_2+1-\varepsilon_2)}{2}$. Since $\varepsilon_2 = 0$, we have $\delta(a, b_1, b_2, b_3) = (S'_1 - S'_2)/2 - \varepsilon_1/2 = \delta(x_2, b_1, b_2, b_3) - \varepsilon_1/2 \leq \delta(x_2, b_1, b_2, b_3)$. However, this case contradicts the hypothesis that $\delta(a, b_1, b_2, b_3) > \delta(x_2, b_1, b_2, b_3)$.
- Else, $S'_3 > S'_2$. Since $S_2 \geq S_3$, we have $\varepsilon_3 = 1$, which implies $S_2 = S_3$. This, in turn, implies that $\delta(x_2, b_1, b_2, b_3) = (S'_1-S'_3)/2 = (S_1-1+\varepsilon_1-S_3)/2 = (S_1-S_2)/2-(1-\varepsilon_1)/2 = \delta(a, b_1, b_2, b_3) - (1 - \varepsilon_1)/2 \leq \delta(a, b_1, b_2, b_3)$.

  In such a case, $\delta(x_2, b_1, b_2, b_3) < \delta(a, b_1, b_2, b_3)$ if and only if we have $\varepsilon_1 = 0$, *i.e.*, x_2 is on a shortest ab_1-path.

□

The metric property of Lemma 9 is illustrated with Figure 8. We use it to strengthen our results in Section 3.2 as follows.

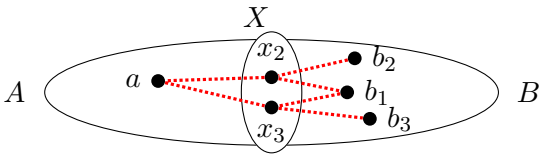


Figure 8: An illustration of the metric property of Lemma 9. The dashed lines represent shortest paths.

Lemma 10. *Let a, b, c, d be a 4-tuple of a connected graph G, and X_a, X_d be two cliques of G satisfying:*

- *X_a is a $(a|b,c,d)$-separator;*
- *X_d is a $(d|a,b,c)$-separator;*
- *all vertices of $X_a \backslash X_d$ and a, b, c are in the same connected component of $G \setminus X_d$.*

Then there exist $x_a \in X_a$ and $x_d \in X_d$ such that

$$\delta(a,b,c,d) \leq \delta(x_a,b,c,x_d) + 1/2.$$

Proof. Consider the three sums $S_1,\ S_2,\ S_3$ from Definition 1. For ease of reasoning, let us order the sums by decreasing value, *i.e.*, let $T_1,\ T_2,\ T_3$ be such that $\{T_1,T_2,T_3\} = \{S_1,S_2,S_3\}$ and $T_1 \geq T_2 \geq T_3$. Accordingly, let u_1,u_2,u_3 be such that $\{u_1,u_2,u_3\} = \{b,c,d\}$ and for every $i \in \{1,2,3\}$, $T_i = \mathrm{d}(a,u_i) + \mathrm{d}(u_j,u_k)$ where $\{j,k\} = \{1,2,3\} \setminus \{i\}$. We distinguish between two cases:

- Suppose there is $x_a \in X_a$ satisfying $\delta(a,b,c,d) \leq \delta(x_a,b,c,d)$. By the hypothesis, the clique X_d is a $(d|x_a,b,c)$-separator. Hence by Lemma 4 there exists $x_d \in X_d$ such that $\delta(x_a,b,c,d) \leq \delta(x_a,b,c,x_d) + \frac{1}{2}$. Altogether, $\delta(a,b,c,d) \leq \delta(x_a,b,c,d) \leq \delta(x_a,b,c,x_d) + \frac{1}{2}$, so, the lemma holds true in this case.

- Else, no vertex of X_a satisfies the above property. In particular, let $x_a \in X_a$ be such that $\mathrm{d}(a,x_a) = \mathrm{d}(a,X_a)$, and x_a is on a shortest au_2-path. By the hypothesis, the clique X_a is a $(a|b,c,d)$-separator. Hence, we can deduce the following information on the 4-tuple:

  - by Lemma 4 $\delta(a,b,c,d) \leq \delta(x_a,b,c,d) + \frac{1}{2}$, and so, $\delta(a,b,c,d) = \delta(x_a,b,c,d) + \frac{1}{2}$;
  - moreover, by Lemma 9 we have that $T_1 > T_2 = T_3$ and x_a is also on a shortest au_1-path.

  We shall prove that there exists $x_d \in X_d$ such that $\delta(x_a,b,c,d) \leq \delta(x_a,b,c,x_d)$, that will prove the lemma in this case.

  For every i, let $T'_i = \mathrm{d}(x_a,u_i) + \mathrm{d}(u_j,u_k)$ where $\{j,k\} = \{1,2,3\} \setminus \{i\}$. Observe that when x_a is on a shortest au_i-path, we have that $T'_i = T_i - \mathrm{d}(a,x_a) = T_i - \mathrm{d}(a,X_a)$. As a result,

  $$T'_1 = T_1 - \mathrm{d}(a,X_a),\ T'_2 = T_2 - \mathrm{d}(a,X_a) \text{ and } T'_3 = T_3 - \mathrm{d}(a,X_a) + 1.$$

  Indeed, the two first equalities follow from the fact that x_a is on a shortest au_1-path, resp. on a shortest au_2-path. The third equality follows from the fact that $\delta(x_a,b,c,d) = (T'_1 - \max\{T'_2,T'_3\})/2 = \delta(a,b,c,d) - 1/2$. In particular, we have $T'_1 \geq T'_3 > T'_2$.

  Furthermore, by the hypothesis X_d is a $(d|x_a,b,c)$-separator. Let v_1,v_2,v_3 be such that $\{v_1,v_2,v_3\} = \{x_a,b,c\}$ and for every i, $T'_i = \mathrm{d}(v_i,d) + \mathrm{d}(v_j,v_k)$ where $\{j,k\} = \{1,2,3\} \setminus \{i\}$.

Finally, let $x_d \in X_d$ be a vertex satisfying $\mathrm{d}(d, x_d) = \mathrm{d}(d, X_d)$ and x_d is on a shortest v_3d-path. We claim that $\delta(x_a, b, c, d) \leq \delta(x_a, b, c, x_d)$, that will prove the lemma in this case. Indeed, suppose for the sake of contradiction that $\delta(x_a, b, c, x_d) < \delta(x_a, b, c, d)$. By Lemma 9, it implies that $T'_1 > T'_3 = T'_2$, which contradicts the fact that $T'_3 > T'_2$.

As a result, in both cases there exist $x_a \in X_a$ and $x_d \in X_d$ such that $\delta(a, b, c, d) \leq \delta(x_a, b, c, x_d) + 1/2$. □

Corollary 11. *Let a, b, c, d be a 4-tuple of a connected graph G satisfying $\delta(a, b, c, d) \geq 3/2$. There exists an atom A of G such that $\delta(a, b, c, d) \leq \delta(G[A]) + 1$.*

Proof. The atom A is obtained by applying Lemma 6. For every vertex $u \in \{a, b, c, d\} \setminus A$, let $X_u \subseteq A$ be a clique-separator disconnecting u from $\{a, b, c, d\} \setminus \{u\}$.

We claim that all vertices of $A \setminus X_u$ and $\{a, b, c, d\} \setminus \{u\}$ are in the same connected component of $G \setminus X_u$. Indeed, since A is an atom, all vertices of $A \setminus X_u$ are in the same connected component C of $G \setminus X_u$. Suppose for the sake of contradiction that there exists $v \in \{a, b, c, d\} \setminus \{u\}$ such that $v \notin C$. Since $v \notin A$, there exists a clique $X_v \subseteq A$ which separates v from $\{a, b, c, d\} \setminus \{v\}$, and so, no vertex of $\{a, b, c, d\} \setminus \{v\}$ is in the same connected component of $G \setminus X_u$ as v. However, in this situation let C_u, C_v be the respective components of u and v in $G \setminus X_u$ and let $X = X_u,\ A = C_u \cup C_v \cup X,\ B = V \setminus (C_u \cup C_v)$. The 4-tuple a, b, c, d is a $(a_1, a_2|b_1, b_2)$ 4-tuple with $a_1 = u,\ a_2 = v$, and so, it implies by Corollary 3 that $\delta(a, b, c, d) \leq 1$, a contradiction. As a result, all vertices of $\{a, b, c, d\} \setminus \{u\}$ are in C, that proves the claim.

We can now find representants for the elements of $\{a, b, c, d\}$ not contained in A. We start by considering the two vertices a and b.

- If $a, b \in A$ then we set $a' = a,\ b' = b$. In this situation $\delta(a, b, c, d) \leq \delta(a', b', c, d)$.
- Else, if $a \in A$ and $b \notin A$ then let $x_b \in X_b$ be a vertex satisfying $\mathrm{d}(b, x_b) = \mathrm{d}(b, X_b)$. Let $a' = a,\ b' = x_b$. By Lemma 4, we have $\delta(a, b, c, d) \leq \delta(a', b', c, d) + 1/2$. In the same way, if $a \notin A,\ b \in A$ then we set $a' = x_a,\ b' = b$ where $x_a \in X_a$ is a vertex satisfying $\mathrm{d}(a, x_a) = \mathrm{d}(a, X_a)$.
- Else, $a, b \notin A$. By the claim above, all vertices of $X_a \setminus X_b \subseteq A \setminus X_b$ and a, c, d are in the same connected component of $G \setminus X_b$. Hence, by Lemma 10 there exist $x_a \in X_a$ and $x_b \in X_b$ such that $\delta(a, b, c, d) \leq \delta(x_a, x_b, c, d) + 1/2$. We set $a' = x_a,\ b' = x_b$ in this case.

Overall, in all cases there exist $a', b' \in A$ such that $\delta(a, b, c, d) \leq \delta(a', b', c, d) + 1/2$. We then proceed similarly with c, d in order to find two vertices $c', d' \in A$ such that $\delta(a', b', c, d) \leq \delta(a', b', c', d') + 1/2$. Altogether, $\delta(a, b, c, d) \leq \delta(a', b', c', d') + 1 \leq \delta(G[A]) + 1$. □

This yields the following approximation result, which only depends upon the hyperbolicity of each individual atom.

Theorem 12. *Let $A_1, \ldots, A_l$ be the atoms of a connected graph G. Then:*

$$\max_i \delta(G[A_i]) \leq \delta(G) \leq \max_i \delta(G[A_i]) + 1.$$

Proof. As for Theorem 5, the lower-bound of Theorem 12 follows from the fact that the subgraphs $G_i = G[A_i]$ are isometric subgraphs of G. The upper-bound trivially holds when $\delta(G) \leq 1$. We can thus suppose that $\delta(G) \geq 3/2$ and so, that there exist four vertices a, b, c, d such that $\delta(a, b, c, d) = \delta(G) \geq 3/2$. Corollary 11 then yields an atom A such that $\delta(G) \leq \delta(G[A]) + 1$, which proves the second part of our claim. □

Note that the upper-bound is reached by the graph of Figure 7, and by the 1-hyperbolic chordal graph from Figure 1 whose atoms have hyperbolicity 0.

5 Substitution method for an exact computation

As shown with Theorem 12 and Fig. 7, the hyperbolicity of a graph cannot be deduced exactly from the hyperbolicity of its atoms. In order to close this gap, we define supergraphs of the atoms (i.e. *substitute graphs*) whose hyperbolicity is more representative.

In this section, we consider graphs with hyperbolicity at least 1. We show that under this assumption, the hyperbolicity of a graph can be computed from the hyperbolicity of the atoms' substitutes.

Outline of the method We build upon Lemma 8 and other results from the previous sections. We recall that a *simplicial* vertex is a vertex whose neighborhood induces a complete subgraph. Given a $(A|B)$-clique separator, we add simplicial vertices to the induced subgraphs $G[A]$ and $G[B]$ in order to mimic the maximum $(a|b_1, b_2, b_3)$ 4-tuples that may result from the disconnection (Section 5.1.1). Since the atoms result from the disconnection of the graph by some of its clique-separators, we can repeatedly apply this method and so obtain the atoms' substitutes. We finally focus on technical details and additional data structures related to the implementation.

5.1 Substitute graphs

In this section we present *substitute graphs* and their properties, in order to show their role in the computation of the hyperbolicity. The algorithmic aspects are covered in Section 5.2.

5.1.1 Substitution around a single $(A|B)$-clique separator

Following [9], the clique-decomposition of a graph G can be computed by repeatedly applying a simple decomposition step, until none of the subgraphs considered contains a clique-separator. The decomposition of G is at first initialized to $\mathcal{G} = \{G\}$. Then, while there exists a graph in $\mathcal{G}$ that contains a clique-separator, we choose arbitrarily one such graph $G' \in \mathcal{G}$ and we proceed as follows. We pick an $(A|B)$-clique separator X such that $A \cap B = X,\ A \cup B = V(G')$, $G'[A]$ is connected and $N_{G'}(A \backslash X) = N_{G'}(B \backslash X) = X$. Then, we remove G' from $\mathcal{G}$ and we add $G'[A]$ and $G'[B]$ to replace it. This decomposition step is slightly modified to yield *substitute graphs* instead of the atoms themselves:

- Let $G_A = G'[A]$. For every $b \in B \setminus X$, we consider the set of vertices $X_b \subseteq X$ which are at distance $\mathrm{d}_{G'}(b, X)$ from b. For every X_b, we add in G_A a (simplicial) vertex s_{X_b} whose neighborhood is X_b. The resulting graph is named G_A^*.
- Let $G_B = G'[B]$. For every $a \in A \setminus X$, we consider the set of vertices $X_a \subseteq X$ which are at distance $\mathrm{d}_{G'}(a, X)$ from a. For every X_a, we add in G_B a (simplicial) vertex s_{X_a} whose neighborhood is X_a. The resulting graph is named G_B^*.

More formally, the *substitute graphs* (or *substitutes* for short) G_A^* and G_B^* of the graphs G_A and G_B with respect to the $(A|B)$-separator X are defined as follows:

Definition 13. *Let X be an $(A|B)$-clique-separator of a connected graph G', where $A \cap B = X$ and $A \cup B = V(G')$. The* substitute graphs *G_A^*, G_B^* are defined as:*

$$V(G_A^*) = A \cup \{s_{X_b} : \exists b \in B \text{ s.t. } X_b = \arg\min_{x \in X} \mathrm{d}(b, x)\} \text{ and } E(G_A^*) = E(A) \cup \{\{s_{X_b}, x\} : x \in X_b\};$$

$$V(G_B^*) = B \cup \{s_{X_a} : \exists a \in A \text{ s.t. } X_a = \arg\min_{x \in X} \mathrm{d}(a, x)\} \text{ and } E(G_B^*) = E(B) \cup \{\{s_{X_a}, x\} : x \in X_a\}.$$

Note that all vertices of $V(G_A^*)\backslash A$ (resp. $V(G_B^*)\backslash B$) have distinct neighborhoods in G_A^* (resp. G_B^*).

Lemma 14. *Let X be a $(A|B)$-clique-separator of a connected graph G', where $A \cap B = X$ and $A \cup B = V(G')$. Suppose $\delta(G') \geq 1$. We have:*

$$\delta(G') = \max\{1, \delta(G_A^*), \delta(G_B^*)\}.$$

Proof. Let $G_A = G'[A]$, $G_B = G'[B]$. By construction (Definition 13), G_A is an isometric subgraph of G_A^*, resp. G_B is an isometric subgraph of G_B^*.

We claim $\delta(G') \leq \max\{1, \delta(G_A^*), \delta(G_B^*)\}$. In order to prove the claim, let a, b, c, d be a 4-tuple of vertices of G' such that $\delta(a, b, c, d) = \delta(G')$. We distinguish between three cases.

1. **Case $a, b, c, d \in A$.** In this situation, $\delta(G') \leq \delta(G_A) \leq \delta(G_A^*)$.

   Similarly, when $a, b, c, d \in B$ we have $\delta(G') \leq \delta(G_B) \leq \delta(G_B^*)$.

2. **Case $a \in A$, $b, c, d \in B$.** The 4-tuple is a $(a|b_1, b_2, b_3)$ 4-tuple. In such case, there exists by construction a simplicial vertex a^* of $V(G_B^*) \setminus B$ that is adjacent to $\{x \in X : \mathrm{d}(a, x) = \mathrm{d}(a, X)\}$. Therefore, by Lemma 8 $\delta(G') \leq \delta(a^*, b_1, b_2, b_3) \leq \delta(G_B^*)$.

   In the same way, if $b \in B$ and $a, c, d \in A$ then $\delta(G') \leq \delta(G_A^*)$.

3. Else, it can be assumed w.l.o.g. that the 4-tuple is a $(a_1, a_2|b_1, b_2)$ 4-tuple. By Corollary 3, $\delta(G') \leq 1$ in this case.

Altogether, $\delta(G') \leq \max\{1, \delta(G_A^*), \delta(G_B^*)\}$, as desired.

Conversely, we claim $\delta(G') \geq \max\{1, \delta(G_A^*), \delta(G_B^*)\}$. By the hypothesis, $\delta(G') \geq 1$. Furthermore, let a, b, c, d be a 4-tuple of G_B^* such that $\delta(a, b, c, d) = \delta(G_B^*)$ (the proof for G_A^* is symmetrical to this one). We distinguish between three cases.

1. **Case $a, b, c, d \in B$.** In this situation, $\delta(G_B^*) \leq \delta(G_B) \leq \delta(G')$.

2. **Case $a = a^* \notin B$ and $b, c, d \in B$.** By construction, a^* is the substitute of some vertex of A, *i.e.*, there exists $a' \in A$ such that $N(a^*) = \{x \in X : \mathrm{d}(a', x) = \mathrm{d}(a', X)\}$. Furthermore, by Lemma 8 $\delta(a^*, b, c, d) = \delta(a', b, c, d)$. Hence, $\delta(G_B^*) \leq \delta(a', b, c, d) \leq \delta(G')$.

3. Else, there are at least two vertices of the 4-tuple that are not in B. Say w.l.o.g. $a, b \notin B$ and let $A^* = X \cup \{a, b\}$, $B^* = V(G_B^*) \setminus \{a, b\}$. In this situation, the 4-tuple is a $(a_1, a_2|b_1, b_2)$ 4-tuple with $a_1 = a$, $a_2 = b$. As a result, by Corollary 3, $\delta(G_B^*) \leq 1 \leq \delta(G')$ in this case.

Altogether, $\delta(G_B^*) \leq \delta(G')$. □

We emphasize that some simple rules can be applied to reduce the size of the substitute graphs, e.g. by removing the pendant vertices which may be added by the construction. We postpone a short analysis of the size of substitutes to Section 5.2.2.

5.1.2 Substitutes of atoms

The substitution operation can be naturally extended to the whole clique-decomposition, by mimicking each step of it and applying the basic substitution operation that we describe above at each of these steps. We formalize it by first introducing the following definition of an atom tree.

Definition 15 ([8, 9, 51])**.** *Let G be a connected graph. An* atom tree *of G is a labeled binary rooted tree T, satisfying the following recursive definition:*

- *if G is prime w.r.t. clique-decomposition, then T is reduced to a node labeled with V;*
- *otherwise, the root of T is labeled with a clique-separator X, and there exists two connected components C_1, C_2 of $G \setminus X$ satisfying:*
  - *$N_G(C_1) = N_G(C_2) = X$;*
  - *the left child of the root is labeled with $A_1 = C_1 \cup X$, which does not contain any clique-separator;*
  - *and the right child of the root is an atom tree of $G \setminus C_1$.*

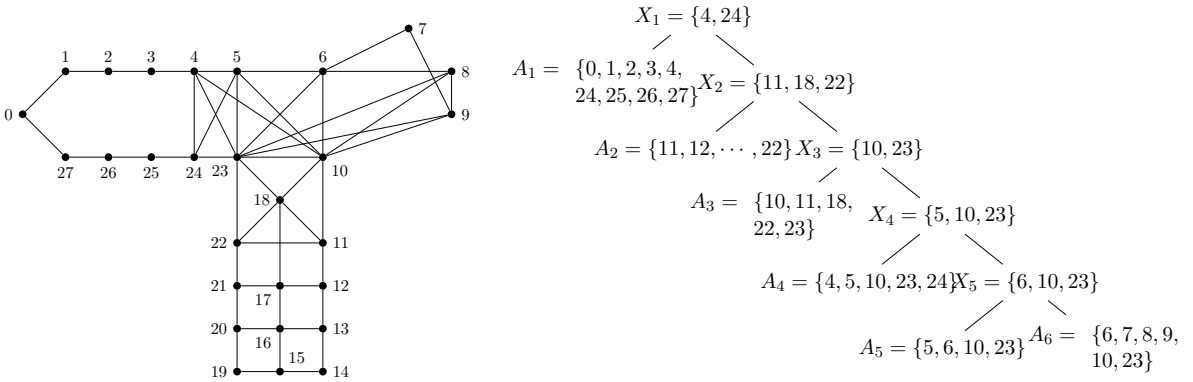
In order to prevent any confusion, the reader has to notice that an atom tree is not necessarily a tree-decomposition (as defined in Section 4.1). In fact, an atom tree can be seen as the trace of some execution of the algorithm of [51, 65] for computing the clique-decomposition. Indeed, it is proved in [51] that in an atom tree, the leaves are in bijective correspondance with the atoms of the graph. Given a *fixed* atom tree, this yields a natural total ordering of the atoms by increasing depth. We now follow this ordering to construct the substitutes of the atoms from the atom tree. There are as many steps in our substitution method as there are atoms in the graph.

- Starting from $H_1 = G$, we disconnect the first atom A_1 by using the clique-separator X_1 from the atom tree. Applying the substitution operation of Definition 13 to $A = A_1$ and $B = V(G) \setminus (A_1 \setminus X_1)$, we obtain two substitute graphs: $G^*_{A_1}$ which substitutes A_1, and another one denoted by $H_2 = G^*_B$.
- After $i - 1$ steps, $i \in \{2, \ldots, l-1\}$, we constructed the substitute graphs of atoms $A_1, \ldots, A_{i-1}$, plus an additional graph H_i. The graph H_i contains $G[\bigcup_{j \geq i} A_j]$, to which were added simplicial vertices during the previous steps. By using the clique-separator X_i from the atom tree we disconnect the graph H_i and we apply the substitution operation of Definition 13, this time to the set A equal to $C_A \cup X_i$ where C_A is the connected component of $H_i \setminus X_i$ which intersects A_i, and to $B = V(H_i) \setminus (A \setminus X_i)$. We replace H_i with the two substitute graphs, one containing the atom A_i and being its substitute, the other being denoted by $H_{i+1} = G^*_B$.
- We finally stop at the l^{th} step, and we set H_l as the substitute graph of the *last* atom A_l.

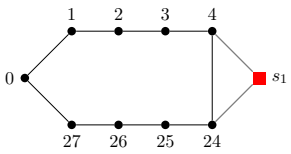
Figure 9 illustrates this process. The numbers reported in Tab. 9i illustrate the interest of our pre-processing method for the computation of the hyperbolicity. Indeed, the graph G of Figure 9a has 28 nodes and so, 20 475 4-tuples, while the sum of the numbers of 4-tuples in the graphs G^*_i (Figs. 9c–9h) is 1 800. We thus significantly reduce the size of the search space. Moreover, a simple cutting rule allows us to reduce the number of 4-tuples to consider to 1 575. To do so, we first order the graphs G^*_i by decreasing diameters, then we iteratively compute the hyperbolicity of these graphs in this order, and we stop exploring as soon as the diameter of a graph G^*_j is smaller than twice the largest value of δ computed so far, given that $2\delta(G^*_j) \leq \mathrm{diam}(G^*_j)$.

5.2 Implementation and complexity analysis

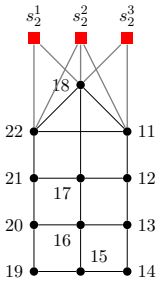
We now explain how to efficiently compute the atoms' substitutes defined in Section 5.1.2. In order to stay competitive with clique-decomposition, the complexity of computing the substitutes



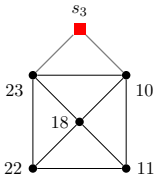
(a) G (b) Atom tree of G



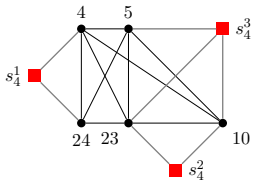
(c) G_1^* built from A_1



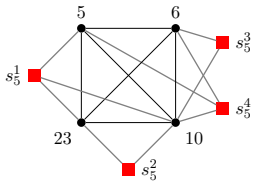
(d) G_2^* built from A_2



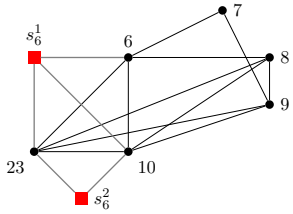
(e) G_3^* built from A_3



(f) G_4^* built from A_4



(g) G_5^* built from A_5



(h) G_6^* built from A_6

	n	m	D	δ	4-tuples
G	28	50	9	2	20 475
G_1^*	10	11	5	2	210
G_2^*	15	25	5	2	1 365
G_3^*	6	10	2	1	15
G_4^*	8	16	3	1/2	70
G_5^*	8	16	2	1/2	70
G_6^*	8	16	3	1	70

(i) Main characteristics

Figure 9: A connected graph G (Figure 9a), an atom tree of the graph (Figure 9b), the substitute of the atoms of G (Figs. 9c–9h), and the characteristics of these graphs (Tab. 9i).

needs to be of the same order of magnitude as the one of computing the atoms. A straightforward calculation would require the computation of all-pairs shortest-paths in the graphs $H_1, H_2, \ldots, H_l$ (defined in Section 5.1.2), hence the computation of l all-pairs shortest-paths. Furthermore, the addition of simplicial vertices at each step causes an increase of the size of the graphs, which further complicates the complexity analysis.

We prove that the atoms' substitutes can all be computed in $\mathcal{O}(nm)$-time (Corollary 20). This is performed through standard partition refinement techniques and using additional properties of clique-decomposition. On the way, we introduce a few rules in order to reduce the size of the substitutes.

5.2.1 Precomputation step and updates

We first focus on some computational tasks that have to be repeatedly executed at each step of our substitution method. In this section, we provide a high-level description for their implementation.

Computation of the distances Given an $(A|B)$-separator X in a graph G, we need to compute all distances $\mathrm{d}(x, v)$ for $x \in X, v \notin X$, in order to compute the substitutes G^*_A, G^*_B . If the distance matrix of G is precomputed then each distance $\mathrm{d}(x, v)$ can be accessed in constant-time, hence an $O(n|X|)$-time complexity. However, the construction from Section 5.1.2 repeatedly introduces new vertices in the substitute graphs: we wish to avoid recomputing the distances from scratch at each step of the substitution method, as that would result in an $\Omega(n^2m)$-time complexity.

Lemma 16. *Let G be a connected graph. We can embed in quadratic time the distance matrix of G into a data structure, supporting:*

- $\mathcal{O}(1)$ *access to the distance between a non-simplicial vertex and any other;*
- $\mathcal{O}(|A|)$ *updates when G is replaced with the substitute G^*_B (w.r.t. a $(A|B)$-clique-separator X).*

Proof. The gist of such a structure is Lemma 8. Let X be a $(A|B)$-clique separator of G, and s be a simplicial vertex added to the substitute graph of $G[B]$. Let $a \in A \setminus X$ satisfy $N(s) = \{x \in X : \mathrm{d}_G(a, x) = \mathrm{d}_G(a, X)\}$. Then we have for every $b \in B$, $\mathrm{d}(b, s) = \mathrm{d}_G(a, b) - \mathrm{d}_G(a, X) + 1$.

It thus follows that once the substitution of a with s has been completed, we only need to remember the association of s with a and an offset $\mathrm{d}(a, X_a)$, so that we can compute the distances in the substitute graphs. More formally, after an arbitrary number of steps of the substitution method, every vertex v' in the graph G' considered is associated with a pair (u, ℓ), with u being a vertex of the original graph G and ℓ being an offset, defined such that $\mathrm{d}_{G'}(v', w) = \mathrm{d}_G(u, w) - \ell$ for every vertex $w \in V(G') \cap V(G)$. At first, when $G' = G$, we only need to store the trivial associations $(v, 0)$ for every $v \in V(G)$. Then, at any further step, once the substitution of a given vertex a with s has been completed, we can compute the association for s in constant-time as follows. We pick any neighbour $x \in X$ of the simplicial vertex s and then we associate s with $(x, \mathrm{d}(a, x) - 1) = (x, \mathrm{d}(a, X) - 1)$. Finally, since there are $l = O(n)$ steps for our substitution method, and that no more than $O(n)$ new simplicial vertices are added at each step, a quadratic-size array is sufficient to store all the pairs $(a, \mathrm{d}(a, X_a) - 1)$. □

Note that the data structure of Lemma 16 does not support the computation of distances between two vertices added by our construction. We can safely ignore this drawback, as we do not need to compute such distances in our method.

Computation of connected components Other complexity bottlenecks arise from the computation of connected components in graphs with a superlinear number of edges (up to $\Omega(nm)$ edges). Indeed, at each step i, $1 \leq i < l$, we need to compute the connected component C_A containing the next atom A_i to deal with. Determining the connected components of a graph is linear-time computable. However, as we detailed in Section 5.1.2, here we have to extract the component from a graph $H_i \neq G$, possibly containing more edges than G due to the addition of simplicial vertices at previous steps. Thus it may result in an $\Omega(m)$-time complexity by using the classical algorithm for this problem. Instead, we propose a method to construct the component incrementally, starting from A_i and adding simplicial vertices at every step $1 \leq j \leq i - 1$.

Lemma 17. *Let G be a connected graph, T be an atom tree of G, and $A_1, \ldots, A_l$ be its atoms ordered according to their depth in T. We denote by $H_1 = G, H_2, \ldots, H_l$ the sequence of l graphs that are computed by our process, each H_i being decomposed into H_{i+1} and the substitute graph of the i^{th} atom by applying the substitution method of Definition 13. For every (simplicial) vertex $s_i \in H_{i+1} \setminus H_i$, we can compute the index j such that s_i belongs to the substitute graph of the j^{th} atom, in total $O(n|X_i|)$-time.*

Proof. Let $s_i \in H_{i+1} \setminus H_i$ be a simplicial vertex. By construction (Section 5.1.2), we have $N(s_i) \subseteq X_i \subseteq V(G)$. Therefore, if s_i belongs to the substitute graph of the j^{th} atom, $j > i$, then it holds that $N(s_i)$ intersects the connected component containing $A_j \setminus X_j$ —in the graph $H_j \setminus X_j$—. In such case since every vertex in the component either belongs to the atom or is simplicial and not in $V(G)$, then it follows that s_i has a neighbor in $A_j \setminus X_j$. Conversely, if $s_i \in V(H_j)$ and $N(s_i) \cap (A_j \setminus X_j) \neq \emptyset$, then s_i is in the same connected component as $A_j \setminus X_j$ in the graph $H_j \setminus X_j$, hence s_i belongs to the substitute graph of the j^{th} atom. So, at every step of our substitution method, if a simplicial vertex is added by our construction, we consider the minimum index j such that $A_j \setminus X_j$ contains a neighbor of the vertex, and we update the vertex set of the substitute graph of the j^{th} atom by adding this new vertex into it. Since only $O(n)$ vertices are added at step i, and that their neighborhood is contained into X_i, the $O(n|X_i|)$-time complexity follows. □

The two above routines (Lemmas 16 and 17) are combined in what follows in order to obtain the desired $\mathcal{O}(nm)$-time complexity for our substitution method. Before this, we show how to reduce the number of simplicial vertices to be added at each step (Section 5.1.1).

5.2.2 Applying simplification rules

The purpose of this section is to reduce the size of the substitutes. Precisely, given $G = (V, E)$ and X a $(A|B)$-clique-separator, we wish to construct a substitute G^*_B from $G[B]$ by adding as few simplicial vertices as possible. A naive implementation would consist in computing the subsets $X_a = \{x \in X : \mathrm{d}_G(a, x) = \mathrm{d}_G(a, X)\}$, for every $a \in A \setminus X$, then adding a simplicial vertex adjacent to X_a. However, by doing so we would add $|A \setminus X|$ new vertices in the substitute, and so, we would lose all the benefit of the separation in terms of size of the graphs. We now define rules in order to avoid this worst-case in some situations. The goal of this section is to give hints on an efficient way to implement these rules, which are of practical interest.

Partition refinement techniques We remove pendant and *twin vertices*. Indeed, it may happen that $X_a = \{x_a\}$ for some $a \in A$, and in such a case we needn't add a simplicial vertex for a since the removal of a pendant vertex does not affect the value of hyperbolicity. Furthermore, it may also happen that $X_a = X_{a'}$ for some pair $a, a' \in A$, and in such a case we wish to add only one simplicial vertex in G^*_B.

To do that efficiently, we use the well-known *partition refinement* techniques (see *e.g.* [42, 56]). Given a partition $\mathcal{P}$ of a set V, and a subset $S \subseteq V$ called the *pivot*, the partition refinement of $\mathcal{P}$ w.r.t. S consists in replacing every group V_i of $\mathcal{P}$ by the non-empty groups among $V_i \cap S$ and $V_i \cap \bar{S}$. This can be achieved in $O(|S|)$-time, up to the precomputation of an appropriate data structure in linear $O(|V|)$-time.

We deduce from this standard technique the following result:

Lemma 18. *Let G be a connected graph given by its distance matrix, and $X \subseteq V(G)$. We define the relation $\equiv_X$ over the set $V(G) \setminus X$ as*

$$u \equiv_X v \iff^{def} \{x \in X : \mathrm{d}_G(u, x) = \mathrm{d}_G(u, X)\} = \{x \in X : \mathrm{d}_G(v, x) = \mathrm{d}_G(v, X)\}.$$

The equivalence classes of $\equiv_X$ can be computed in $O(n|X|)$-time.

Proof. Since the distance matrix is given, we can compute $X_u = \{x \in X \mid \mathrm{d}_G(u, x) = \mathrm{d}_G(u, X)\}$ for every vertex u, that takes $\mathcal{O}(n|X|)$-time. Then, we start from the partition $\mathcal{P} = \{V \setminus X\}$ which we refine successively for every $x \in X$ with the set $\{u : u \in V \setminus X \text{ s.t. } \mathrm{d}_G(u, x) = \mathrm{d}_G(u, X)\}$. The total cost is $O(\sum_{x \in X} |N_{G_X}(x)|) = O(n|X|)$. □

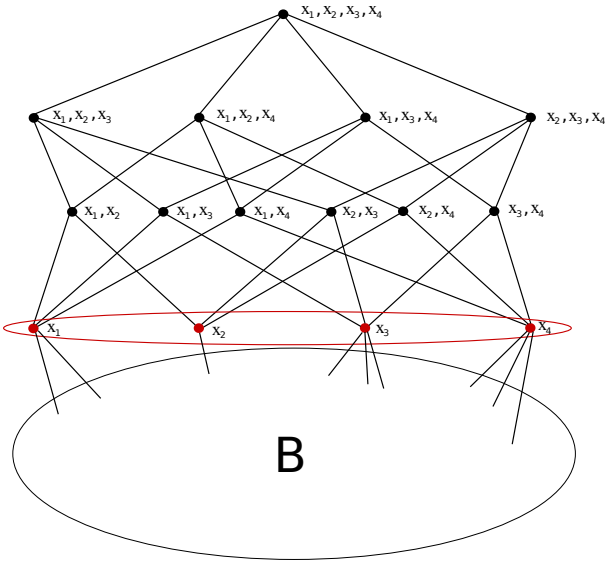


Figure 10: A case when $|A \setminus X| = 2^{|X|} - |X| - 1$ vertices need to be added to G^*_B. The graph $G[A]$ is obtained from the subset lattice of the set $X = \{x_1, x_2, x_3, x_4\}$. Every vertex $a \in A$ is labeled with $\{x \in X \mid \mathrm{d}(a, x) = \mathrm{d}(a, X)\}$.

There are at most $2^{|X|}$ equivalence classes of $\equiv_X$, and we can remove the $|X| + 1$ classes corresponding to the subsets of X of size at most one. Altogether, the substitute G^*_B has at most $|B| + \min\{|A \setminus X|, 2^{|X|} - |X| - 1\}$ vertices when we apply our simplification rules. This bound is sharp, as shown with Figure 10.

Overall, the substitution method adds at most $\sum_{i=1}^{l-1} \min\{n, 2^{|X_i|} - |X_i| - 1\}$ new vertices to the vertex-set of the atoms. In particular, consider the special case when all but at most c clique-separators X_i have size at most k, for some universal constants c and k. In this situation, there are at most $(c + 2^k - k - 1) \cdot n$ new vertices added to the atoms. Hence, the total number of vertices is only increased by a constant factor. This property holds for outerplanar graphs and some real-life graphs that we study in the next two sections.

5.2.3 Complexity analysis

Finally, to determine the time complexity of our substitution method, we use the following result:

Lemma 19 ([8])**.** *Let G be a connected graph, and $A_1, \ldots, A_l$ be its atoms. Then $\sum_i |A_i| \leq n + m$.*

Corollary 20. *The substitute of the atoms of a connected graph G can be computed in $O(nm)$-time.*

Proof. The notations are the same as for Section 5.1.2. That is, fix any atom tree T of G and let $A_1, \ldots, A_l$ be the atoms ordered by increasing depth. For every $i < l$, let X_i be the clique-separator of G labeling the father node of leaf A_i in T. By Definition 15, $X_i \subseteq A_i$.

We first precompute the distance matrix of G in $O(nm)$-time, then we embed it in quadratic-time into the data structure of Lemma 16. Also, for every i, we initialize the vertex set of the i^{th} substitute graph with the atom A_i. We then apply each step of our substitution method sequentially.

Precisely, at each step i we are given the vertex set V_i of the graph H_i to be considered. Initially, $V_1 = V(G)$.

- Let us partition the vertices in A and B, where A is the set of all vertices in V_i that are in the i^{th} substitute graph. Since A is known (initialized with A_i then updated at each previous step), it can be done in $\mathcal{O}(|V_i|)$-time.
- Then, we compute the simplicial vertices that result from the substitution operation, and we add them in A and B, respectively. Since we have constant-time access to the distances, we have by Lemma 18 that it can be done in $\mathcal{O}(|V_i||X_i|)$-time.
- For every vertex newly added at this step, we next compute the index of the atom's substitute to which it belongs to. By Lemma 17, it can also be done in $\mathcal{O}(|V_i||X_i|)$-time.
- We set $V_{i+1} = B$ and then we update the distances accordingly. By Lemma 17, it can be done in $\mathcal{O}(|A|)$-time, that is $\mathcal{O}(|V_i|)$.

We can easily show by induction that $|V_i| = \mathcal{O}(n)$. Hence, the i^{th} step can be executed in $\mathcal{O}(n|X_i|)$-time. Overall, our modified clique-decomposition can be computed in $\mathcal{O}(n\sum_i |X_i|)$-time, that is in $\mathcal{O}(n\sum_i |A_i|) = \mathcal{O}(nm)$-time by Lemma 19. □

6 Hyperbolicity of outerplanar graphs

Equipped with the substitution method of Section 5 and our in-depth analysis of clique-decomposition (Sections 3 and 4), we aim at applying these results in order to speed-up the computation of hyperbolicity in some graph classes. In the next two sections, we review theoretical and practical cases when it is possible to do so. We start with a linear-time algorithm computing the hyperbolicity of a given outerplanar graph (Theorem 28).

Outerplanar graphs can be characterized in several ways (see [62]). A *planar graph* is a graph drawable in the Euclidean plane so that edges may only intersect at their endpoints. It is *outerplanar* if it stays planar whenever one adds a universal vertex to it. Equivalently, a graph is outerplanar if it is drawable in the Euclidean plane so that edges may only intersect at their endpoints, and all the vertices lie on a common face which is called the *outerface*. Such a drawing is furthermore called an outerplanar embedding, and it can be computed in linear time [63]. Note that it easily follows from this definition that all cycles are outerplanar graphs.

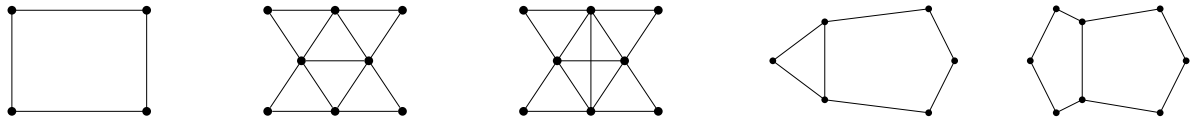
The class of outerplanar graphs is minor-closed, and a graph is outerplanar if and only if it is K_4-minor-free and $K_{2,3}$-minor-free [62].

We exploit nontrivial properties of the clique-decomposition of outerplanar graphs to prove Theorem 28. In particular, minimal clique-separators of an outerplanar graph have size at most two, *i.e.*, they are either cut-vertices or edge-separators. More precisely, the atoms of an outerplanar graph are cycles. We use the following formula, which yields the hyperbolicity of a given cycle:

Lemma 21 ([69, 25])**.** *Cycles of order $4p+\varepsilon \geq 3$, with $p \geq 0$ and $\varepsilon \in \{0,1,2,3\}$, are $(p-1/2)$-hyperbolic when $\varepsilon = 1$, and p-hyperbolic otherwise.*

The main purpose of this section is to obtain a similar characterization for the substitute of cycles (Lemma 26). So, we analyze the properties of this class of graphs in Section 6.2. We also need to compute the substitutes of atoms of a given outerplanar graph in linear time. For this purpose, we rely on the notion of *weak dual* [5].

Definition 22. *Let G be a biconnected outerplanar graph. The weak dual of G is a tree T_G equal to the intersection graph of the atoms of G. Two adjacent nodes of T_G correspond to atoms which share a single edge.*



(a) C_4 (b) H_1 (c) H_2 (d) H_3 (e) H_4

Figure 11: Characterization of 5-chordal $\frac{1}{2}$-hyperbolic graphs, in terms of forbidden isometric subgraphs.

Note that a weak dual is nothing else than a tree-decomposition whose bags are the atoms of an outerplanar graph. If G is biconnected, then starting from an outerplanar embedding of the graph, we construct it by removing from the dual of the graph the universal vertex corresponding to the outerface (see Figure 13 for an example).

We show that the atoms' substitutes can be computed by dynamic programming on the weak dual (Lemma 27). This combined with Lemma 26 and a characterization of 1/2-hyperbolic outerplanar graphs (Proposition 23) achieves proving Theorem 28.

6.1 Outerplanar graphs with small hyperbolicity

As observed in Section 5, our substitution method for an exact computation of hyperbolicity requires the hyperbolicity of the graph to be at least 1. To overcome this drawback, we first characterize in this section outerplanar graphs that are $\frac{1}{2}$-hyperbolic. Note that we only consider biconnected graphs, as the hyperbolicity of a graph is the maximum hyperbolicity taken over all its biconnected components, and the biconnected components of a graph are computable in linear time [64].

Proposition 23. *A biconnected outerplanar graph is $\frac{1}{2}$-hyperbolic if, and only if, either it is isomorphic to C_5, or it is chordal and it does not contain the graph of Figure 11b as a subgraph. Furthermore, these conditions can be checked in linear time.*

Proof. Let G be a $\frac{1}{2}$-hyperbolic outerplanar biconnected graph. By Lemma 21, the graph C_5 is $\frac{1}{2}$-hyperbolic, and we now assume that G is not isomorphic to C_5. The induced cycles of G are exactly its atoms. As a result, we have by Lemma 21 that G only has induced cycles of length 3 or 5 (else, $\delta(G) \geq 1$). Moreover, Wu and Zhang prove in [69] that a 5-chordal graph is $\frac{1}{2}$-hyperbolic if, and only if, it does not contain any graph of Figure 11 as an isometric subgraph[3].

Since we consider graphs with induced cycles of length 3 or 5, G is C_4-free and so, it does not contain the graph of Figure 11a as an isometric subgraph. Moreover, we claim that G is C_5-free, as otherwise it would contain the graph of Figure 11d, or the graph of Figure 11e, as an isometric subgraph. Thus G has to be chordal. In addition, we claim that G cannot contain the graph of Figure 11c, since it is not outerplanar and being outerplanar is a hereditary property. Consequently, G is $\frac{1}{2}$-hyperbolic if, and only if, it is chordal and it does not contain the graph of Figure 11b as an isometric subgraph.

Let H_1 be the graph of Figure 11b. To complete the proof of the proposition, we are left to prove that every subgraph of G that is isomorphic to H_1 is isometric. Indeed, the latter will prove that G is $\frac{1}{2}$-hyperbolic if, and only if, it is chordal and it does not contain H_1 as a subgraph.

We observe that H_1 is an edge-maximal outerplanar graph, hence every subgraph isomorphic to H_1 must be induced. Suppose for the sake of contradiction that there is an induced subgraph

[3]The characterization of [69] is composed of six forbidden isometric subgraphs, but the sixth one is actually 6-chordal.

of G that is isomorphic to H_1 but not isometric. In this situation, there is a vertex $x \in V(G \setminus H_1)$ connecting two vertices at distance 3 in H_1. As shown in Figure 12, it implies the existence of a $K_{2,3}$-minor in G, thereby contradicting that it is outerplanar. Therefore, the claim is proved, and so, G is $\frac{1}{2}$-hyperbolic if, and only if, it is chordal and it does not contain H_1 as a subgraph.

Being chordal can be checked in linear time [57]. Furthermore, when G is chordal, all its induced cycles have length three, hence it contains H_1 as a subgraph if and only if there are two adjacent vertices of degree 3 in its weak dual (see Figure 13 for an illustration). Overall, since the weak dual can be computed in linear time, deciding whether a chordal outerplanar graph G is 1/2-hyperbolic can be done in linear time. □

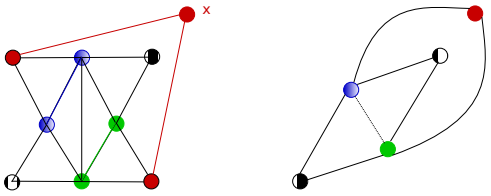


Figure 12: Existence of a $K_{2,3}$-minor in G.

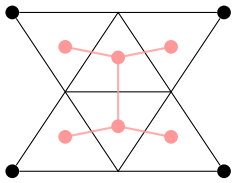


Figure 13: The forbidden subgraph of Figure 11b, and its characterization in the weak dual.

6.2 Substitute graphs of cycles

As we constrain ourselves to outerplanar graphs, recall that the atoms are exactly the induced cycles of the graph. Clearly, a clique-separator contained in a cycle is either a cut-vertex or an edge-separator. Since we never add pendant vertices with our substitution method (cf. Section 5.2.2), we never add a simplicial vertex in the first case, and in the second case we might only add a single vertex which has to be adjacent to both ends of the edge-separator. Substitute graphs of cycles thus fall into the following subclass of outerplanar graphs:

Definition 24. *A biconnected outerplanar graph is called a sunshine graph if it can be obtained from a cycle C by adding, for every edge of C, at most one simplicial vertex that is adjacent to both endpoints of that edge.*

All cycles are sunshine graphs, and two other examples of sunshine graphs are given in Figure 14. We can derive some useful properties of sunshine graphs from their definition. First is that vertices in the cycle C form a dominating set of the graph. Furthermore, the choice of C is unique, except for the particular case of the diamond graph (obtained from two triangles sharing an edge).

Finally, since every sunshine graph G is obtained by adding simplicial vertices to a cycle, it has at most one induced cycle of length at least four, and if it exists this cycle must be C. Thus, we have by Theorem 12 that $\delta(C) \leq \delta(G) \leq \delta(C) + 1$. This difference can be decreased by $\frac{1}{2}$ as follows.

Lemma 25. *Let G be a sunshine graph, and C be a dominating cycle of G. Then we have:*

$$\delta(C) \leq \delta(G) \leq \delta(C) + \frac{1}{2}.$$

Proof. By Theorem 12, we have $\delta(C) \leq \delta(G) \leq \delta(C)+1$. So, we are left to prove that, for each 4-tuple a, b, c, d, we have $\delta(a, b, c, d) \leq \delta(C) + 1/2$. We consider five different cases:

1. Suppose that all the nodes a, b, c, d belong to C. Then, since C is an isometric cycle of G, it is trivial that $\delta(a, b, c, d) \leq \delta(C) \leq \delta(C) + 1/2$.

2. Suppose that three nodes among a, b, c, d belong to C. W.l.o.g., $a \notin C$ but $b, c, d \in C$. Let $X = N(a)$, $A = N[a]$, and $B = V(G) \setminus a$. The 4-tuple is a $(a|b_1, b_2, b_3)$ 4-tuple, and so, by Lemma 4, there exists $x \in N(a)$ such that $\delta(a, b, c, d) \leq \delta(x, b, c, d) + 1/2$. In this situation, $x, b, c, d \in C$, hence $\delta(a, b, c, d) \leq \delta(C) + 1/2$.

3. Suppose that two nodes belong to C. In the same way as above, we can assume w.l.o.g. that $a, d \notin C$ but $b, c \in C$. Let $X_a = N(a)$ and $X_d = N(d)$. These two clique-separators satisfy the conditions of Lemma 10, so, there exist $x_a \in X_a$, $x_d \in X_d$ such that $\delta(a, b, c, d) \leq \delta(x_a, b, c, x_d) + 1/2$. In this situation, $x_a, b, c, x_d \in C$, hence $\delta(a, b, c, d) \leq \delta(C) + 1/2$.

4. Suppose only one node belongs to C. We can assume w.l.o.g. that $a \in C$. Furthermore, let us arbitrarily orient the cycle C. For every $u \in \{b, c, d\}$, we denote by $e_u = \{h_u, t_u\}$ the edge of C induced by its neighbors, where h_u denotes the head of the edge w.r.t. the orientation. Observe that for every $u, v \in \{b, c, d\}$, we have $\mathrm{d}(u, v) = 2 + \min\{\mathrm{d}(h_u, t_v), \mathrm{d}(h_v, t_u)\} = 1 + \mathrm{d}(h_u, h_v) = 1 + \mathrm{d}(t_u, t_v)$. In particular, for every $u \in \{b, c, d\}$ we have $\mathrm{d}(a, h_u) \leq \mathrm{d}(a, u) \leq \mathrm{d}(a, h_u) + 1$.

   Let S_1, S_2, S_3 satisfy $\{S_1, S_2, S_3\} = \{\mathrm{d}(a, b) + \mathrm{d}(c, d), \mathrm{d}(a, c) + \mathrm{d}(b, d), \mathrm{d}(a, d) + \mathrm{d}(b, c)\}$ and $S_1 \geq S_2 \geq S_3$. Accordingly, let u_1, u_2, u_3 satisfy $\{u_1, u_2, u_3\} = \{b, c, d\}$ and for every i, $S_i = \mathrm{d}(a, u_i) + \mathrm{d}(u_j, u_k)$ where $\{j, k\} = \{1, 2, 3\} \setminus i$. Finally, let $S'_i = \mathrm{d}(a, h_{u_i}) + \mathrm{d}(h_{u_j}, h_{u_k})$.

   We have $\mathrm{d}(a, h_{u_i}) \leq \mathrm{d}(a, u_i) \leq \mathrm{d}(a, h_{u_i}) + 1$ and $\mathrm{d}(u_j, u_k) = \mathrm{d}(h_{u_j}, h_{u_k}) + 1$. Consequently, we have $S'_i + 1 \leq S_i \leq S'_i + 2$ for every i. By the 4-point condition, it implies $\delta(a, h_b, h_c, h_d) \leq \delta(a, b, c, d) + 1/2$. Since in addition $a, h_b, h_c, h_d \in C$, we have $\delta(a, b, c, d) \leq \delta(C) + 1/2$.

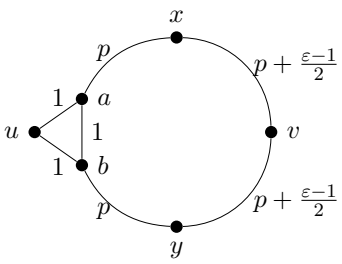
5. Finally, suppose that no node belongs to C. As in the previous case, we arbitrarily orient the cycle C and for every $u \in \{a, b, c, d\}$ we denote by $e_u = \{h_u, t_u\}$ the edge of C induced by its neighbors, where h_u denotes the head of the edge w.r.t. the orientation. In this situation, for every $u, v \in \{a, b, c, d\}$, we have $\mathrm{d}(u, v) = 1 + \mathrm{d}(h_u, h_v)$. Therefore by the 4-point condition we have $\delta(a, b, c, d) = \delta(h_a, h_b, h_c, h_d) \leq \delta(C) \leq \delta(C) + 1/2$.

Altogether, this proves $\delta(G) \leq \delta(C) + 1/2$. □

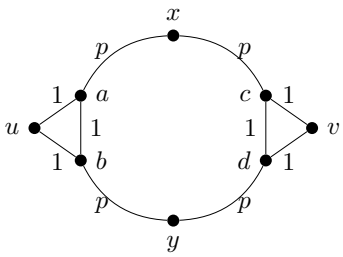
We now present a characterization of the hyperbolicity of sunshine graphs, from which one can easily derive a linear-time algorithm to compute it.

Lemma 26. *Let G be a sunshine graph, and C be a dominating cycle for G of length $4p+\varepsilon \geq 3$, with $p \geq 0$ and $\varepsilon \in \{0, 1, 2, 3\}$. Assuming $G \setminus C$ is nonempty we have:*

- *if $\varepsilon \in \{1, 3\}$, then $\delta(G) = \delta(C) + \frac{1}{2}$;*
- *if $\varepsilon = 2$, then $\delta(G) = \delta(C) + \frac{1}{2}$ if there is a diametral pair made of two simplicial vertices not in C, and $\delta(G) = \delta(C)$ otherwise;*



(a) n odd



(b) $n = 4p + 2$

Figure 14: Substitute graphs of the atoms of an outerplanar graph.

- *if* $\varepsilon = 0$*, then* $\delta(G) = \delta(C)$.

Proof. Recall that by the previous Lemma 26, we have $\delta(G) \leq \delta(C) + \frac{1}{2}$. Thus we only focus on finding 4-tuples u, v, x, y of hyperbolicity (at least) this value, and we choose one, if any, maximizing $|C \cap \{u, v, x, y\}|$. In what follows, write $S_1 = \mathrm{d}(u, v) + \mathrm{d}(x, y)$, $S_2 = \mathrm{d}(u, x) + \mathrm{d}(v, y)$ and $S_3 = \mathrm{d}(u, y) + \mathrm{d}(v, x)$. We assume in addition that $S_1 \geq S_2 \geq S_3$.

Case ε odd In such a case, we have $\delta(C) = p + \frac{\min\{0, \varepsilon-2\}}{2}$ by Lemma 21. Figure 14a exhibits a 4-tuple u, v, x, y satisfying:

$$\begin{aligned} S_1 &= (2p + \frac{\varepsilon + 1}{2}) + (2p + \min\{1, \varepsilon - 1\}) = 4p + \frac{\varepsilon + 1}{2} + \min\{1, \varepsilon - 1\}; \\ S_2 &= (p + 1) + (p + \frac{\varepsilon - 1}{2}) = 2p + \frac{\varepsilon + 1}{2}; \\ S_3 &= S_2. \end{aligned}$$

Hence, this 4-tuple has hyperbolicity $p + \frac{\min\{1, \varepsilon-1\}}{2} = \delta(C) + \frac{1}{2}$.

Case $\varepsilon = 2$ In such a case, we have $\delta(C) = p$ by Lemma 21. We assume w.l.o.g. that $u \notin C$, and we claim that it implies $v \notin C$. Indeed, by the metric property of Lemma 9, and noticing that $S_1 \geq S_2 \geq S_3$, the vertex v has to be at equal distance l of both neighbors of u, as otherwise u could be replaced with one of its two neighbors, contradicting the maximality of $|C \cap \{u, v, x, y\}|$. Hence $v \in C$ is impossible, as it would yield the length of C is $2l + 1 = 4p + 2$. It thus follows that $v \notin C$, and the length of C is in fact $2(l - 1) + 2 = 2l$, yielding $l = 2p + 1$.

Conversely, assume that there exist two simplicial vertices u, v that are diametrically opposed in G. We choose the 4-tuple u, v, x, y as in Figure 14b, and it satisfies:

$$\begin{aligned} S_1 &= (2p + 2) + (2p + 1) = 4p + 3; \\ S_2 &= 2(p + 1) = 2p + 2; \\ S_3 &= S_2. \end{aligned}$$

So, we have $\delta(u, v, x, y) = p + \frac{1}{2} = \delta(C) + \frac{1}{2}$.

Case $\varepsilon = 0$ Another application of Lemma 21 yields $\delta(C) = p$. Assuming $u \notin C$, we deduce as for the previous case that $v \notin C$, and v is at equal distance $l = 2p$ from both neighbors of u. Thus, C is partitioned by the neighborhoods of u and v in two paths of length $l - 1 = 2p - 1$, that is in the same way as in Figure 14b. Furthermore, since the diameter of C is $2p$, those paths are geodesics of the cycle.

We recall that on the way to prove Lemma 25, we showed that $u, v, x, y \notin C$ implies $\delta(u, v, x, y) \leq \delta(C)$. Since we assume $\delta(u, v, x, y) > \delta(C)$, it implies the existence of one vertex $z \in \{x, y\}$ among the 4-tuple that must be in C. Furthermore, in this situation we obtain by considering the geodesic containing z that $\mathrm{d}(u, z) + \mathrm{d}(v, z) = 2 + (l - 1) = 2p + 1$. In particular, we have $\min\{\mathrm{d}(u, z), \mathrm{d}(v, z)\} \leq p$. The latter contradicts the assumption that $\delta(u, v, x, y) > \delta(C)$, since we have by [59] that $\delta(u, v, x, y) \leq \min\{\mathrm{d}(u, z), \mathrm{d}(v, z)\}$. Altogether, we always have $\delta(G) = \delta(C)$ if $\varepsilon = 0$. □

6.3 Applying the substitution method in linear time

Given an outerplanar graph G, we recall that we aim at computing $\delta(G)$ from the substitute of its atoms. From Lemma 26, the hyperbolicity of the substitutes can be computed in linear time. So, we are left to prove that the atom's substitutes can also be computed in linear time.

Lemma 27. *Let G be an outerplanar biconnected graph. The substitute graphs of the atoms of G can be computed in linear time.*

Proof. We construct the weak dual T_G of G from an outerplanar embedding, that is linear-time computable. Let $C_1, \ldots, C_l$ be the atoms of G. We root T_G on an atom C_1, which is an induced cycle. Then, we claim that the following algorithm for computing the atom's substitutes is correct.

- For every i, we initialize C_i^* with C_i.
- We start a depth-first search from the root, and so obtain a postordering of the nodes of T_G. Then, we visit the atoms following this ordering, and we proceed as follows. For every C_i, we name $e_{ij} = C_i \cap C_j$ an edge shared with a child in the rooted tree. If there is a vertex of C_j^* that is at equal distance to both ends of e_{ij} then we add in C_i^* a new simplicial vertex that is adjacent to e_{ij}. That is, we add such new vertex if either C_j is odd, or there is a simplicial vertex of $C_j^* \setminus C_j$ that is adjacent to the edge opposed to e_{ij}.
- Finally, we start a breadth-first search from the root and for every visited atom $C_i \neq C_1$, we consider its parent atom, denoted by C_k, naming $e_{i,k}$ the edge-separator that it shares with it. As before, we add in C_i^* a simplicial vertex whose neighborhood is $e_{i,k}$ if, and only if, either the length of C_k is odd, or there is a simplicial vertex in $C_k^* \setminus C_k$ whose neighborhood is the edge diametrically opposed to $e_{i,k}$ in the atom C_k.

This algorithm runs in linear time. Furthermore, we note that an atom tree can be obtained from $(T_G; C_1)$ (as defined in Definition 15) by disconnecting at every step an atom that is a leaf of T_G until the clique-decomposition is obtained. Following this atom tree, the atom's substitutes so obtained are isomorphic to the output $C_1^*, C_2^*, \ldots, C_l^*$ of the above algorithm. In particular, for every atom, we use the depth-first search to compute the simplicial vertices resulting from the disconnection of its sons, whereas the breadth-first search is used to compute the single vertex resulting from its own disconnection, if any. Hence, the above algorithm for computing the atom's substitutes is correct, and so, the resulting $C_1^*, \ldots, C_l^*$ are the substitute graphs of the atoms of G. □

Figure 15 shows the substitute graphs resulting from the application of the substitution method to a biconnected outerplanar graph.

We finally conclude with the following theorem.

Theorem 28. *The hyperbolicity of a given connected outerplanar graph G is computable in linear time.*

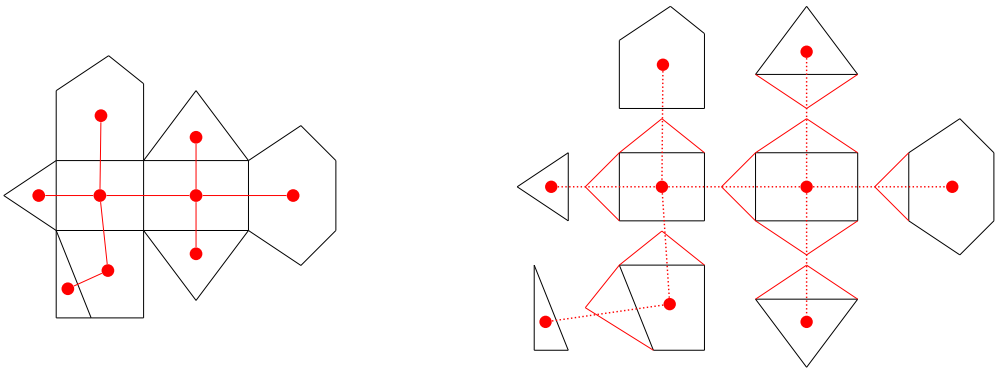


Figure 15: An application of the substitution method to an outerplanar graph.

Proof. We can safely assume G to be biconnected by [64]. By [6, 43], G is 0-hyperbolic if, and only if, G is a clique. If it is not, then by Proposition 23, we can check whether it is $\frac{1}{2}$-hyperbolic in linear time.

From now on, assume $\delta(G) \geq 1$. By Lemma 27, we can compute the substitute graphs of the atoms of G in linear time. We can thus conclude by Lemma 14 (*i.e.* the correctness of our substitution method), as these substitute graphs are sunshine graphs and their hyperbolicity is linear-time computable by Lemma 26. □

7 Experimental evaluation

Before concluding the paper, we shall apply the substitution method of Section 5 to some real-life graphs. We report in this section on experiments performed with our substitution methodology on the graphs of six collaboration networks and social networks and eight biological networks. This way, we aim at evaluating the computation time of the substitutes on some empirical graphs, and to better understand the factors impacting their size (compared with the upper-bound of Section 5.2.2).

The section is subdivided as follows. In Section 7.1, we present the graphs from the dataset and we motivate our choice to test the method on these graphs. We report on the reduction on the size of the subgraphs (biconnected components, atoms and substitutes) in Section 7.2. In spite of strong similarities between the graphs from the dataset, the results obtained vary from one graph to the other. So, we conduct a deeper analysis of their clique-decomposition in Section 7.3, reporting on the structure of their atom tree and on the size of the clique-separators, in order to justify the variations in the results. We complete our experiments with a numerical analysis of the time needed for computing the hyperbolicity of these graphs, with and without the clique-decomposition and the substitute decomposition of Section 5. On the way, we report on the hyperbolicity of all graphs in the dataset (Section 7.4).

7.1 Datasets

Collaboration networks We apply the algorithm presented in Section 5 to the collaboration networks of five different scientific communities [52], namely:

- `ca-AstroPh`, for the astrophysics community;
- `ca-CondMat`, for the condensed matter physics community;
- `ca-GrQc`, for the general relativity and quantum cosmology community;

- `ca-HepPh`, for the high energy physics-phenomenology community;
- and `ca-HepTh`, for the high energy physics-theory community.

In the `ca-*` graphs, nodes represent scientists and edges represent collaborations (i.e., co-authoring a paper). These graphs are interesting to analyze the behavior of our algorithm, and the size of their substitute graphs, because they have many cliques of various sizes. Indeed, a paper co-authored by k scientists induces a clique of size k in the graph. Furthermore, the number of co-authors per papers varies from one community to another. As noted in Section 5.2.2, there are $\mathcal{O}(2^k)$ vertices newly added to the substitutes for any clique-separator of size k. Therefore, we expect to observe different results in terms of the size of the substitutes, despite the graphs from the dataset share many properties (see [52]).

Biological networks We also consider biological networks from the Biological General Repository for Interaction Datasets (BioGRID [61, 21], version 3.4.145) and the Database of Interacting Proteins [58]. The complete list of the networks considered is available in Table 3. These graphs represent interactions (*e.g.*, protein-protein interaction networks) and associated annotation based on Experimental System. We expect their distribution of clique-separators to be correlated with the size of the interactions (number of entities involved).

Social networks We finally use the graph `soc-loc-brightkite` [24] representing friendship relations in the location-based social network of the service provider Brightkite. Users shared their locations by checking-in. Furthermore, since social graphs have a high clustering coefficient [68], we expect to observe a significant decrease of the size of this graph using our substitution method.

Our results are summarized in Table 3. In what follows, we only detail the results obtained for collaboration networks. A detailed analysis of our results for the other types of networks is left for specialists.

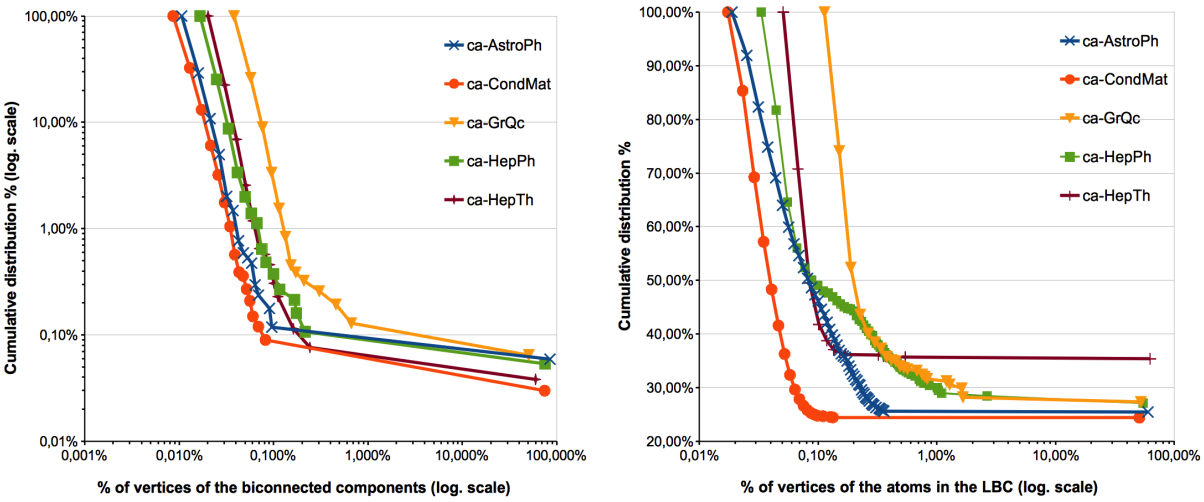
7.2 Empirical results

We modified the clique-decomposition algorithm of [9] to implement the substitution method that we presented in Section 5. We used it here to compute, for every graph, the substitute of each atom of the decomposition.

Below, we report on the size of the substitutes. We compare it with the size of the atoms and the biconnected components (see Figure 16 and Table 1). This preliminary analysis also explains why we can ignore almost all substitutes in the computation of hyperbolicity (precisely, all but one substitute), that further reduces the time of computation.

Decomposition into biconnected components We observed that all of the five graphs are composed of one largest biconnected component, that we call LBC, that includes from 50% to 84.85% of all the vertices. This can be observed from the cumulative distribution of the size of the biconnected components in Figure 16a. The cumulative number of components is given as a percentage of the total number of biconnected components, and the size of the components as a percentage of the total number of vertices in the graph. We noticed that all the biconnected components but the LBC are small: only covering at most 1% of the vertices.

Clearly, the smallest biconnected components can be safely ignored for the computation of hyperbolicity, provided that their diameter is smaller than two times the hyperbolicity of the LBC, which is always the case for these graphs (see [25]). Thus, we now focus on the clique-decomposition of the LBC, and on its resulting substitute graphs.



(a) Biconnected components

(b) Atoms in the LBC

Figure 16: Cumulative distributions of the size of the biconnected components (Figure 16a) and of the atoms in the LBC of each graph (Figure 16b).

Clique-decomposition We plotted in Figure 16b the cumulative distribution of the size of the atoms of the LBC. The cumulative number of atoms is given as a percentage of the total number of atoms and the size of the atoms as a percentage of the total number of vertices in the LBC. Again, for all of the graphs, we observed one largest atom, that we call the LA, that includes from 50% to 60% of all the vertices, and all the other atoms only represent a small fraction of the overall vertices.

Moreover, like for the smallest biconnected components and as reported in [25], the substitute graphs of the smallest atoms can be safely ignored for the computation of hyperbolicity. As a result, the only component of the graphs to deal with for computing their hyperbolicity is the substitute graph of the LA. We denote it by LS in what follows.

Size of the substitute graphs As explained in Section 5, the size of the LS depends on both the initial size of the LA and the number of added simplicial vertices. We have reported in Table 1 the original size n of each graph, the size n_B of its LBC, the size n_{LA} of the LA, and the size n_{LS} of the largest substitute. We have then computed the percentage R_{LA} of vertices that have been removed from the LBC to obtain the LA, that is $R_{LA} = \frac{n_B - n_A}{n_B}$. We observe a significant reduction rate R_{LA}, varying from 37.40% to 49.22%. We have also computed the reduction rate R_{LS} of the LS with respect to the LBC, that is $R_{LS} = \frac{n_B - n_{LS}}{n_B}$. We observe that this reduction rate falls between 11.22% and 20.84%. It indicates that in spite of the simplification rules presented in Section 5.2.2, the substitution method adds many simplicial vertices to the LA when constructing the LS.

We reported in Table 1 as *Cost* the percentage of vertices in the LBC representing the addition of new simplicial vertices.

We first analyze the `ca-CondMat` graph. This graph has the largest reduction rates R_{LS} and R_{LA} from the dataset. However, despite a R_{LA} of 49.22%, it has almost the same reduction rate R_{LS} as `ca-HepPh` and `ca-GrQc` — ranging from 20.55% to 20.84%. This is the consequence of more simplicial vertices added with our substitution methodology. The new simplicial vertices represent 28.39% of the size of its LBC, whereas for the `ca-HepPh` graph it goes up to only 24.88%.

Instance name	n	n_B	n_{LA}	n_{LS}	R_{LA}	R_{LS}	Cost	Time (in sec.)
ca-CondMat	23 133	17 234	8 751	13 643	49.22%	20.84%	28.39%	672
ca-GrQc	5 242	2 651	1 386	2 107	47.72%	20.52%	27.20%	5
ca-HepPh	12 008	9 025	4 925	7 170	45.43%	20.55%	24.88%	167
ca-AstroPh	18 772	15 929	9 561	13 407	39.98%	15.83%	24.14%	679
ca-HepTh	9 877	5 898	3 692	5 236	37.40%	11.22%	26.18%	53

Table 1: Characteristics of the collaboration networks. The size of the graph is given as n, the size of the LBC as n_b, the size of the LA as n_{LA} and the size of the LS as n_{LS}. The percentage of vertices removed from the LBC to obtain the LA is given as R_{LA}, the reduction rate is $R_{LS} = \frac{n_B - n_{LS}}{n_B}$, and the percentage of vertices in the LBC, representing the addition of simplicial vertices, is given as *Cost*. Finally, the computation time of the substitution method, denoted by $Time$, is given in seconds.

A similar behavior is observed between ca-AstroPh and ca-CondMat: even though their R_{LA} differ on 9.25%, the difference of their reduction rate R_{LS} finally falls to 5%. This results from the addition of 4.24% less simplicial vertices in ca-AstroPh than in ca-CondMat. As an extremal case, the R_{LA} and R_{LS} of the ca-HepPh graph are respectively bigger and smaller than the R_{LA} and R_{LS} of ca-GrQc.

We thus conclude that the impact of n_{LA} and of the number of new simplicial vertices on the final size n_{LS} differs greatly depending on the graph.

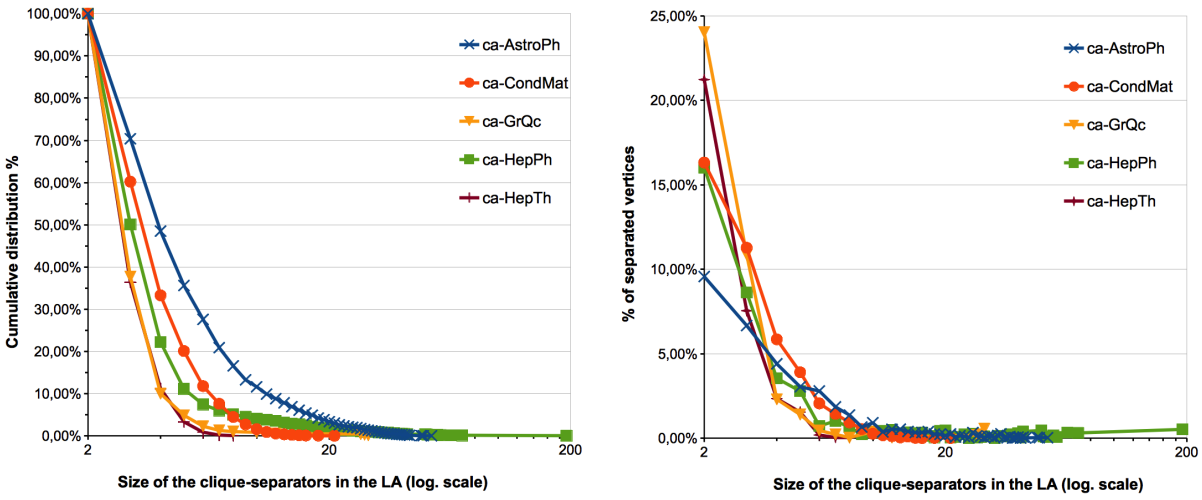
7.3 Decomposition analysis

Having noticed the heterogeneous results of our empirical section, we are now analyzing in more details the properties causing the asymmetry between the various ca-* graphs. To do so, we report on the structure of the intersection graph of the atoms (sometimes call the atom graph) and on the size of the clique-separators.

We support through our experiments that most clique-separators are small (with no more than two or three nodes), and they are responsible for the largest part of new simplicial vertices. One plausible explanation for small-size separators are the existence of people who only published one paper (*e.g.*, student interns publishing one paper with their supervisors before changing their lab or leaving the community; PhD students who only published their thesis; people publishing by mere chance; etc.).

Clique-decomposition We first analyzed the neighborhood of the LA in the *atom graph*, as it is defined in [8]. That is, we consider the set of atoms $\mathcal{A}_{LA} = \{A_1, \ldots, A_l\}$ that intersect the LA, naming $\mathcal{X}_{LA} = \{X_1 = A_1 \cap LA, \ldots, X_l = A_l \cap LA\}$ the clique-separators at their intersection. We emphasize that there might be other atoms in the graph than the LA and those in $\mathcal{A}_{LA}$. But such atoms, if any, do not overlap the LA.

We plotted in Figure 17a the cumulative distribution of the size of the clique-separators in the LA as a percentage of the total number of clique-separators. By doing so, we observed smaller clique-separators for the ca-HepTh and ca-CondMat graphs, with a maximum size of 8 and 21, respectively, than for the three other graphs ca-GrQc, ca-AstroPh and ca-HepPh, having clique-separators of maximum size 42, 53 and 192, respectively. Also, we reported in Table 2 that the ratio $R_{|\mathcal{X}_{LA}|} = |\mathcal{X}_{LA}|/n_{LA}$ varies from 0.39 for ca-AstroPh to 0.54 for ca-CondMat. To sum up, there are more clique-separators in ca-CondMat than in ca-AstropPh, but there are larger clique-separators in ca-AstropPh than in ca-CondMat.



(a) Clique-separators in the LA

(b) Separated vertices from the LA

Figure 17: Cumulative distribution of the size of the clique-separators in the LA (Figure 17a) and percentage of separated vertices as a function of the size of the clique-separators in the LA (Figure 17b).

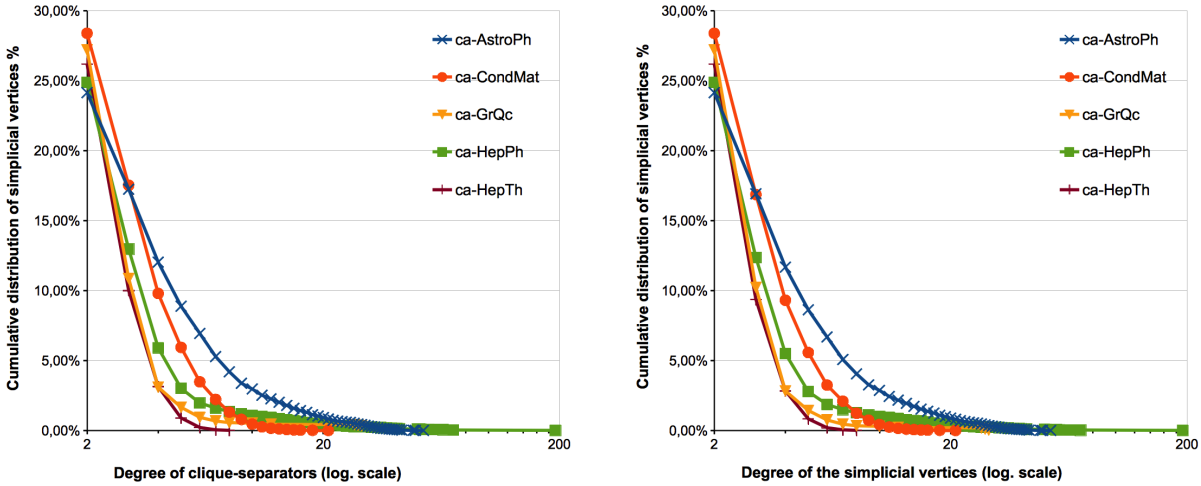
Recall that our substitution methodology never adds more simplicial vertices than the number of nodes disconnected by the clique-separator. So, to complete our measurements, we related the size of clique-separators with the proportion of vertices that are disconnected by them from the LA. We reported in Table 2 as $\alpha_1 = n_B - n_{LA}$ the total number of vertices separated from the LA in the LBC, and as $\alpha_2 = |\bigcup_{i=1}^{l} A_i \setminus X_i|$ the number of vertices of $LBC \setminus LA$ present in an atom of $\mathcal{A}_{LA}$. Finally, we computed the fraction $\Delta_1 = \frac{\alpha_1 - \alpha_2}{n_B}$, quantifying the percentage of vertices that are neither contained into the LA, nor in any of the atoms in $\mathcal{A}_{LA}$. We reported as $\Delta_2 = R_{LA} - \Delta_1$ the fraction of vertices in some atom of $\mathcal{A}_{LA}$, hence those that are *directly* separated from the LA.

Our results put in evidence that most of the vertices are either contained in the LA, or in some other atom intersecting the LA. Other vertices comprise around 2.88% and 7.03% of the overall vertices. Moreover, as shown with Figure 17b, where we plotted the percentage of separated vertices as a function of the size of clique-separators, smaller clique-separators of size ≤ 5 are responsible for a significant part (w.r.t. Δ_2) of the vertices disconnected from the LA in `ca-CondMat` (37.34% of vertices over 49.22%), whereas in `ca-AstroPh` they solely disconnect 23.67% over 39.98% of vertices. This difference is not balanced with clique-separators of larger size, even though these ones disconnect 13.43% of vertices in `ca-AstroPh`, while only 5.67% in `ca-CondMat`. Comparing `ca-CondMat` with `ca-HepPh` does not change the picture. In contrast, for the graphs `ca-GrQc` and `ca-HepTh`, we notice that 6.71% and 4.91% more vertices, respectively, than in `CondMat`, are disconnected by edge-separators. But the rest of the clique-separators only disconnect 16.67% and 11.70% of the vertices from the LA, respectively, whereas 26.70% of them are separated from the LA in `ca-CondMat`. Therefore, most of the difference for the final size of the substitute graph LS comes from the number of vertices that are disconnected by clique-separators of *small* size.

Substitute construction Recall that we assume that the largest number of simplicial vertices are connected to the smallest clique-separators. In order to validate the assumption, we plotted in Figure 18a the cumulative number of simplicial vertices connected to the LA, normalized

Instance name	α_1	α_2	$\lvert\mathcal{X}_{LA}\rvert$	$R_{\mathcal{X}_{LA}}$	$\Delta_1\%$	$\Delta_2\%$
`ca-CondMat`	8 483	7 413	4 702	0.54	6.21%	43.01%
`ca-GrQc`	1 265	1 079	698	0.5	7.03%	40.69%
`ca-HepPh`	4 100	3 727	2 166	0.44	4.13%	41.3%
`ca-AstroPh`	6 368	5 910	3 715	0.39	2.88%	37.1%
`ca-HepTh`	2 206	1 942	1 506	0.41	4.47%	32.93%

Table 2: Distribution of clique-minimal separators, and of the vertices disconnected from the LA.
The total number of vertices separated from the LA in the LBC is given as $\alpha_1 = n_B - n_{LA}$, and the number of disconnected vertices being present in the subset of neighboring atoms $\mathcal{A}_{LA}$ as $\alpha_2 = |V(\mathcal{A}_{LA} \setminus \mathcal{X}_{LA})|$.
Also, the number of clique-minimal separators in the LA is given as $|\mathcal{X}_{LA}|$.
We quantify the percentage of vertices that are neither contained into the LA nor in any of the atoms in $\mathcal{A}_{LA}$ as $\Delta_1 = \frac{\alpha_1 - \alpha_2}{n_B}$; the fraction of vertices in some atom of $\mathcal{A}_{LA}$ that are *directly* separated from the LA is equal to $\Delta_2 = R_{LA} - \Delta_1$.



(a) Simplicial vertices connected to the LA as a function of the size of the clique-separators

(b) Degree distribution of the simplicial vertices connected to the LA

Figure 18: Cumulative number of simplicial vertices connected to the LA normalized by the size of the LBC as a function of the size of the clique-separators to which they are connected (Figure 18a), cumulative degree distribution of the simplicial vertices connected to the LA normalized by the size of the LBC (Figure 18b)

by the size of the LBC, as a function of the size of the clique-separators. In particular, note that for each graph, such a summation is equal to the value given as `Cost` in Table 1. By looking only at clique-separators of size two and three, the proportions of simplicial vertices for the graphs `ca-CondMat`, `ca-GrQc`, `ca-HepPh`, `ca-AstroPh` and `ca-HepTh` respectively, represent 65.49%, 88.63%, 76.26%, 50.16% and 88.02% respectively, of the total number of simplicial vertices connected to the LA. Thus it highlights the importance of clique-separators of small size, to which a large proportion of simplicial vertices are connected to.

Let us also remark by comparing Figure 18a to Figure 18b that almost all simplicial vertices have *same degree*. In the worst case (`ca-GrQc`), there are no more than 0.75% of the simplicial vertices whose degree differs from the others. Most of these simplicial vertices have degree two.

Hence, the final proportion of simplicial vertices, given in Table 1 as `Cost`, mostly depends on the size distribution of the clique-separators in the graphs. Also, since there is a worst-case variation of only 4.25% in the proportion of simplicial vertices in our graphs – that is reached with `ca-CondMat` and `ca-AstroPh` –, that allows us to make relative comparisons between the graphs from the dataset. Especially we are interested in comparing the proportion of simplicial vertices of small degree (less than four). Such a proportion represents, for `ca-CondMat`, `ca-GrQc`, `ca-HepPh`, `ca-AstroPh` and `ca-HepTh` respectively, a percentage of 18.59%, 24.1%, 18.97%, 12.11% and 23.04% respectively, of the simplicial vertices in total. To sum up:

- when comparing `ca-AstroPh` to `ca-CondMat`: even if the former has 2.23% more simplicial vertices with degree at least three, this is compensated by its 6.48% less simplicial vertices of degree at most four, which results in overall to 4.25% less simplicial vertices in `ca-AstroPh` than in `ca-CondMat`. The same happens when comparing `ca-AstroPh` to the remaining graphs. Indeed, the lower number of simplicial vertices with degree at most four *always* compensates its larger number of simplicial vertices of higher degree.

- when comparing `ca-CondMat` to `ca-GrQc` and `ca-HepTh`: the two latter graphs respectively have 5.51% and 4.45% more simplicial vertices with degree at most four. However, they respectively have 6.7% and 6.66% less simplicial vertices with degree at least three. As a result, there are 1.19% less simplicial vertices in `ca-GrQc`, and 2.21% less simplicial vertices in `ca-HepTh`, respectively, than in `ca-CondMat`.

- finally, when comparing `ca-CondMat` to `ca-HepPh`: we observe quite similar numbers of simplicial vertices of degree smaller than four. They respectively represent 18.59% and 18.97% of the simplicial vertices in total. Again, the main difference comes from the proportion of simplicial vertices with degree higher than three, with 5.91% more simplicial vertices in `ca-CondMat` than in `ca-HepPh`, resulting in 3.51% less vertices in `ca-HepPh`.

7.4 Computation times

In order to complete the empirical section, we present in Table 3 the computation times of the hyperbolicity on the LBC and on the LS of the `ca-*` graphs. Of course, we expect the computation time to decrease proportionally to the size of the graphs. However, we use the algorithm proposed in [26] for computing the hyperbolicity, and so, the computation time may be impacted by other factors.

On the way, we also give in Table 3 the values of the hyperbolicity we obtained and the computation time T_{LS} of the LS using the algorithm given in Section 5. Interestingly, for all graphs from the dataset, the hyperbolicity of the graph always equals the hyperbolicity of its largest atom.

We observe that the hyperbolicity of the `ca-*` graphs can be computed from 8 to 24 times faster on the LS than on the LBC. However, computing the hyperbolicity on the LS also comes at the cost of the time to construct this graph. By combining the two, one improves the time of computation by a smaller factor (R_2) between 1.26 and 3.19 (for the graphs `ca-GrQC`, `ca-HepPh` and `ca-CondMat`). In some cases (`ca-AstroPh` and `ca-HepTh`), our method even increases the time of computation. However, these cases happen only for small graphs where the hyperbolicity can be computed in a few seconds.

We have also reported in Table 3 the values of the hyperbolicity and the computation times of the LS and of the hyperbolicity on the LBC and on the LS of some protein-protein interaction networks [61, 21, 58] and of a social network [24]. For these graphs, our method allows for significant reduction of the overall computation times, saving from hours to days of computation on the larger graphs.

Instance name	n	n_B	n_{LA}	n_{LS}	δ	T_{LS}	$T_{\delta_{LS}}$	$T_{\delta_{LBC}}$	R_1	R_2
ca-GrQc	5 242	2 651	1 386	2 107	3.5	6	1	8.8	8.8	1.26
ca-HepTh	9 877	5 898	3 692	5 236	4	52	0.3	2.4	8	0.046
ca-HepPh	12 008	9 025	4 925	7 170	3	162	50	677	13.5	3.19
ca-AstroPh	18 772	15 929	9 561	13 407	3	762	22	202	9.2	0.26
ca-CondMat	23 133	17 234	8 751	13 643	3.5	1180	101	2 498	24.7	1.95
BioGRID SYSTEM [61, 21]										
Dosage Rescue	3 100	1 521	1 335	1 451	4	0.8	3.4	5	1.45	1.16
Synthetic Lethality	4 609	2 258	2 117	2 226	2	1.9	5.7	10.3	1.8	1.36
Synthetic Growth Defect	5 403	3 013	2 598	2 724	2	6	4.3	6	1.43	0.6
Biochemical Activity	7 035	2 944	2 748	2 845	3	4	12	18	1.44	1.08
Affinity Capture RNA	8 734	3 339	2 409	2 480	2	11	0.3	0.5	1.99	0.04
Affinity Capture Western	17 822	9 971	8 732	9 595	4	128	1 873	3 529	1.88	1.76
Affinity Capture MS	31 038	17 793	12 964	13 493	3	491	59 129	150 119	2.54	2.52
BioGRID MV Physical [61, 21]	18 365	9 851	8 674	9 439	4.5	92	1 265	2 658	2.1	1.96
DIP (20170205) [58]	27 029	13 969	13 229	13 675	4.5	255	79 316	90 315	1.14	1.13
soc-loc-brightkite [24]	58 228	33 187	28 792	32 223	3	5 873	5 541	16 985	3.06	1.53

Table 3: Hyperbolicity and computation times on all graphs from the dataset. The size of the graph is given as n, the size of the LBC as n_b, the size of the LA as n_{LA} and the size of the LS as n_{LS}. The value of the hyperbolicity computed on the LS is given as δ. The computation time in second of the largest substitute is given as T_{LS}. The computation times in second of the hyperbolicity, on the LS and on the LBC respectively, is given as $T_{\delta_{LS}}$ and $T_{\delta_{LBC}}$ respectively. Finally, the ratio $T_{\delta_{LBC}}/T_{\delta_{LS}}$ is given as R_1 and $T_{\delta_{LBC}}/(T_{LS} + T_{\delta_{LS}})$ is given as R_2.

8 Conclusion

We proved a tight relationship between the hyperbolicity of a given graph and the maximum hyperbolicity of its atoms. This gives a new proof that chordal graphs (and other related graph classes such as 2-chordal graphs [53, 54], nearly chordal graphs [17] or quasi-triangulated graph [60]) have a bounded hyperbolicity [18]. Our results also cover some class with unbounded hyperbolicity, namely the outerplanar graphs, for which we give a complete characterization of their hyperbolicity. This extends to a linear-time algorithm for computing the hyperbolicity of these graphs. To the best of our knowledge, this is the first linear-time algorithm for computing the hyperbolicity in these graphs. We let open whether the same can be done for other classes of graphs. Especially, can it be taken advantage of the linear-time algorithm for outerplanar graphs in order to compute the hyperbolicity of planar graphs more efficiently?

Furthermore, we deduced from our proofs a general substitution method, allowing us to modify the atoms at no extra-cost than the clique-decomposition. For graphs with hyperbolicity at least one, the maximum hyperbolicity from the resulting graphs is exactly the hyperbolicity of the graphs. However, the graphs to be considered may have a larger size than the atoms. Experiments suggest that the final size of the substitute graphs is mostly related to the number of clique-separators of small size, and the disconnections resulting from them. Part of our future work will consist in finding other graph decompositions which are applicable to the computation of the hyperbolicity.

References

[1] M. Abu-Ata and F. F. Dragan. Metric tree-like structures in real-life networks: an empirical study. *Networks*, 67(1):49–69, 2016.

[2] R. Albert, B. DasGupta, and N. Mobasheri. Topological implications of negative curvature for biological and social networks. *Physical Review E*, 89(3):032811, 2014.

[3] H. Alrasheed and F. F. Dragan. Core-periphery models for graphs based on their δ-hyperbolicity: An example using biological networks. In *CompleNet VI*, pages 65–77. 2015.

[4] R. Anstee and M. Farber. On bridged graphs and cop-win graphs. *Journal of Combinatorial Theory, Series B*, 44(1):22–28, 1988.

[5] B. Baker. Approximation algorithms for NP-complete problems on planar graphs. *Journal of the ACM (JACM)*, 41(1):153–180, 1994.

[6] H.-J. Bandelt and H. Mulder. Distance-hereditary graphs. *Journal of Combinatorial Theory, Series B*, 41(2):182–208, 1986.

[7] S. Bermudo, J. Rodríguez, J. Sigarreta, and J.-M. Vilaire. Gromov hyperbolic graphs. *Discrete Mathematics*, 313(15):1575–1585, 2013.

[8] A. Berry, R. Pogorelcnik, and G. Simonet. Efficient clique decomposition of a graph into its atom graph. Technical Report RR-10-07, LIMOS, Aubière, France, Mar. 2010.

[9] A. Berry, R. Pogorelcnik, and G. Simonet. An introduction to clique minimal separator decomposition. *Algorithms*, 3(2):197–215, 2010.

[10] A. Berry, R. Pogorelcnik, and G. Simonet. Organizing the atoms of the clique separator decomposition into an atom tree. *Discrete Applied Mathematics*, 177:1 – 13, 2014.

[11] H. L. Bodlaender. Discovering treewidth. In *31st Conference on Current Trends in Theory and Practice of Computer Science (SOFSEM)*, volume 3381 of *Lecture Notes in Computer Science*, pages 1–16. Springer, 2005.

[12] M. Boguñá, F. Papadopoulos, and D. V. Krioukov. Sustaining the Internet with hyperbolic mapping. *Nature Communications*, 1(62):1–18, Oct. 2010.

[13] J. A. Bondy and U. Murty. *Graph theory with applications*, volume 290. Macmillan London, 1976.

[14] M. Borassi, A. Chessa, and G. Caldarelli. Hyperbolicity measures democracy in real-world networks. *Physical Review E*, 92(3):032812, 2015.

[15] M. Borassi, D. Coudert, P. Crescenzi, and A. Marino. On computing the hyperbolicity of real-world graphs. In *23rd Annual European Symposium on Algorithms (ESA)*, volume 9294 of *Lecture Notes in Computer Science*, pages 215–226, Patras, Greece, Sept. 2015. Springer.

[16] M. Borassi, P. Crescenzi, and M. Habib. Into the square: On the complexity of some quadratic-time solvable problems. *Electronic Notes in Theoretical Computer Science*, 322:51–67, 2016.

[17] A. Brandstädt and C. T. Hoàng. On clique separators, nearly chordal graphs, and the maximum weight stable set problem. *Theoretical Computer Science*, 389(1):295–306, 2007.

[18] G. Brinkmann, J. H. Koolen, and V. Moulton. On the hyperbolicity of chordal graphs. *Annals of Combinatorics*, 5(1):61–69, 2001.

[19] W. Carballosa, D. Pestana, J. Rodríguez, and J. Sigarreta. Distortion of the hyperbolicity constant of a graph. *the Electronic Journal of Combinatorics*, 19(1):P67, 2012.

[20] J. Chakerian and S. Holmes. Computational tools for evaluating phylogenetic and hierarchical clustering trees. *Journal of Computational and Graphical Statistics*, 21(3):581–599, 2012.

[21] A. Chatr-aryamontri, R. Oughtred, L. Boucher, J. Rust, C. Chang, N. K. Kolas, L. O'Donnell, S. Oster, C. Theesfeld, A. Sellam, C. Stark, B.-J. Breitkreutz, K. Dolinski, and M. Tyers. The BioGRID interaction database: 2017 update. *Nucleic Acids Research*, 45(D1):D369–D379, 2017.

[22] V. Chepoi, F. F. Dragan, B. Estellon, M. Habib, and Y. Vaxès. Diameters, centers, and approximating trees of delta-hyperbolic geodesic spaces and graphs. In *24th Symposium on Computational Geometry (SCG)*, pages 59–68. ACM, 2008.

[23] V. Chepoi, F. F. Dragan, and Y. Vaxes. Core congestion is inherent in hyperbolic networks. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2264–2279. SIAM, 2017.

[24] E. Cho, S. A. Myers, and J. Leskovec. Friendship and mobility: user movement in location-based social networks. In *ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1082–1090, San Diego, CA, 2011. ACM.

[25] N. Cohen, D. Coudert, and A. Lancin. Exact and approximate algorithms for computing the hyperbolicity of large-scale graphs. Rapport de recherche RR-8074, Inria, Sept. 2012.

[26] N. Cohen, D. Coudert, and A. Lancin. On computing the gromov hyperbolicity. *Journal of Experimental Algorithmics (JEA)*, 20(1):1–6, 2015.

[27] D. Coudert and G. Ducoffe. Recognition of C_4-free and 1/2-hyperbolic graphs. *SIAM Journal on Discrete Mathematics*, 28(3):1601–1617, Sept. 2014.

[28] W. H. Cunningham. Decomposition of directed graphs. *SIAM Journal on Algebraic Discrete Methods*, 3(2):214–228, 1982.

[29] B. DasGupta, M. Karpinski, N. Mobasheri, and F. Yahyanejad. Effect of gromov-hyperbolicity parameter on cuts and expansions in graphs and some algorithmic implications. Technical Report arXiv:1510.08779, ArXiv, 2015.

[30] P. de La Harpe and E. Ghys. *Sur les groupes hyperboliques d'après Mikhael Gromov*, volume 83. Progress in Mathematics, 1990.

[31] M. Didi Biha, B. Kaba, M.-J. Meurs, and E. SanJuan. Graph decomposition approaches for terminology graphs. In *MICAI 2007: Advances in Artificial Intelligence*, volume 4827 of *Lecture Notes in Computer Science*, pages 883–893. Springer, 2007.

[32] R. Diestel. *Graph theory, graduate texts in mathematics, vol. 173*. Springer, Heidelberg, 1997.

[33] Y. Dourisboure and C. Gavoille. Tree-decompositions with bags of small diameter. *Discrete Mathematics*, 307(16):2008–2029, 2007.

[34] F. Dragan. Tree-like structures in graphs: A metric point of view. In *39th International Workshop on Graph-Theoretic Concepts in Computer Science (WG)*, volume 8165 of *Lecture Notes in Computer Science*, pages 1–4. Springer, 2013.

[35] A. Dress, K. Huber, J. Koolen, V. Moulton, and A. Spillner. *Basic Phylogenetic Combinatorics*. Cambridge University Press, Cambridge, UK, Dec. 2011.

[36] D. Epstein, M. S. Paterson, J. W. Cannon, D. F. Holt, S. V. Levy, and W. P. Thurston. *Word processing in groups*. AK Peters, Ltd., 1992.

[37] H. Fournier, A. Ismail, and A. Vigneron. Computing the gromov hyperbolicity of a discrete metric space. *Information Processing Letters*, 115(6):576–579, 2015.

[38] J. Gagneur, R. Krause, T. Bouwmeester, and G. Casari. Modular decomposition of protein-protein interaction networks. *Genome Biology*, 5(8):R57, 2004.

[39] T. Gallai. Transitiv orientierbare graphen. *Acta Mathematica Hungarica*, 18(1):25–66, 1967.

[40] A. Goldman. Optimal center location in simple networks. *Transportation science*, 5(2):212–221, 1971.

[41] M. Gromov. Hyperbolic groups. In S. Gersten, editor, *Essays in Group Theory*, volume 8 of *Mathematical Sciences Research Institute Publications*, pages 75–263. Springer, New York, 1987.

[42] M. Habib, C. Paul, and L. Viennot. Partition refinement techniques: An interesting algorithmic tool kit. *International Journal of Foundations of Computer Science*, 10(02):147–170, 1999.

[43] E. Howorka. On metric properties of certain clique graphs. *Journal of Combinatorial Theory, Series B*, 27(1):67–74, 1979.

[44] R. Impagliazzo, R. Paturi, and F. Zane. Which problems have strongly exponential complexity? In *Foundations of Computer Science, 1998. Proceedings. 39th Annual Symposium on*, pages 653–662. IEEE, 1998.

[45] E. A. Jonckheere and P. Lohsoonthorn. Geometry of network security. In *American Control Conference*, volume 2, pages 976–981, Boston, MA, USA, 2004. IEEE.

[46] B. Kaba, N. Pinet, G. Lelandais, A. Sigayret, and A. Berry. Clustering gene expression data using graph separators. *In Silico Biology*, 7(4):433–452, 2007.

[47] W. S. Kennedy, I. Saniee, and O. Narayan. On the hyperbolicity of large-scale networks and its estimation. In *International Conference on Big Data (Big Data)*, pages 3344–3351. IEEE, 2016.

[48] J. H. Koolen and V. Moulton. Hyperbolic bridged graphs. *European Journal of Combinatorics*, 23(6):683–699, 2002.

[49] R. Krauthgamer and J. Lee. Algorithms on negatively curved spaces. In *IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 119–132. IEEE, 2006.

[50] F. Le Gall. Faster algorithms for rectangular matrix multiplication. In *IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 514–523, New Brunswick, NJ, USA, 2012. IEEE.

[51] H.-G. Leimer. Optimal decomposition by clique separators. *Discrete Mathematics*, 113(1):99–123, 1993.

[52] J. Leskovec, J. Kleinberg, and C. Faloutsos. Graph evolution: Densification and shrinking diameters. *ACM Transactions on Knowledge Discovery from Data*, 1(1):1–41, Mar. 2007.

[53] S. McCullough. 2-chordal graphs. In *Contributions to Operator Theory and its Applications*, pages 143–192. Springer, 1988.

[54] S. McCullough. Minimal separators of 2-chordal graphs. *Linear algebra and its applications*, 184:187–199, 1993.

[55] K. Olesen and A. Madsen. Maximal prime subgraph decomposition of bayesian networks. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 32(1):21–31, 2002.

[56] R. Paige and R. Tarjan. Three partition refinement algorithms. *SIAM Journal on Computing*, 16(6):973–989, 1987.

[57] D. J. Rose, R. E. Tarjan, and G. S. Lueker. Algorithmic aspects of vertex elimination on graphs. *SIAM Journal on computing*, 5(2):266–283, 1976.

[58] L. Salwinski, C. S. Miller, A. J. Smith, F. K. Pettit, J. U. Bowie, and D. Eisenberg. The database of interacting proteins: 2004 update. *Nucleic Acids Research*, 32(Database issue):D449–51, 2004.

[59] M. A. Soto Gómez. *Quelques propriétés topologiques des graphes et applications à internet et aux réseaux*. PhD thesis, Univ. Paris Diderot (Paris 7), 2011.

[60] J. P. Spinrad. Recognizing quasi-triangulated graphs. *Discrete applied mathematics*, 138(1):203–213, 2004.

[61] C. Stark, B.-J. Breitkreutz, T. Reguly, L. Boucher, A. Breitkreutz, and M. Tyers. BioGRID: a general repository for interaction datasets. *Nucleic Acids Research*, 34(suppl_1):D535–D539, 2006.

[62] M. Sysło. Characterizations of outerplanar graphs. *Discrete Mathematics*, 26(1):47–53, 1979.

[63] R. Tamassia and I. Tollis. Planar grid embedding in linear time. *IEEE Transactions on Circuits and Systems*, 36(9):1230–1234, 1989.

[64] R. E. Tarjan. Depth-first search and linear graph algorithms. *SIAM Journal on Computing*, 1(2):146–160, 1972.

[65] R. E. Tarjan. Decomposition by clique separators. *Discrete Mathematics*, 55(2):221–232, 1985.

[66] V. Vassilevska Williams, J. Wang, R. Williams, and H. Yu. Finding four-node subgraphs in triangle time. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '15, pages 1671–1680. SIAM, 2015.

[67] K. Verbeek and S. Suri. Metric embedding, hyperbolic space, and social networks. In *Annual Symposium on Computational Geometry (SCG)*, pages 501–510, New York, NY, USA, 2014. ACM.

[68] D. J. Watts and S. H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440–442, 1998.

[69] Y. Wu and C. Zhang. Hyperbolicity and chordality of a graph. *The Electronic Journal of Combinatorics*, 18(1):P43, 2011.

[70] M. Yancey. An investigation into graph curvature's ability to measure congestion in network flow. Technical Report arXiv:1512.01281, ArXiv, 2015.